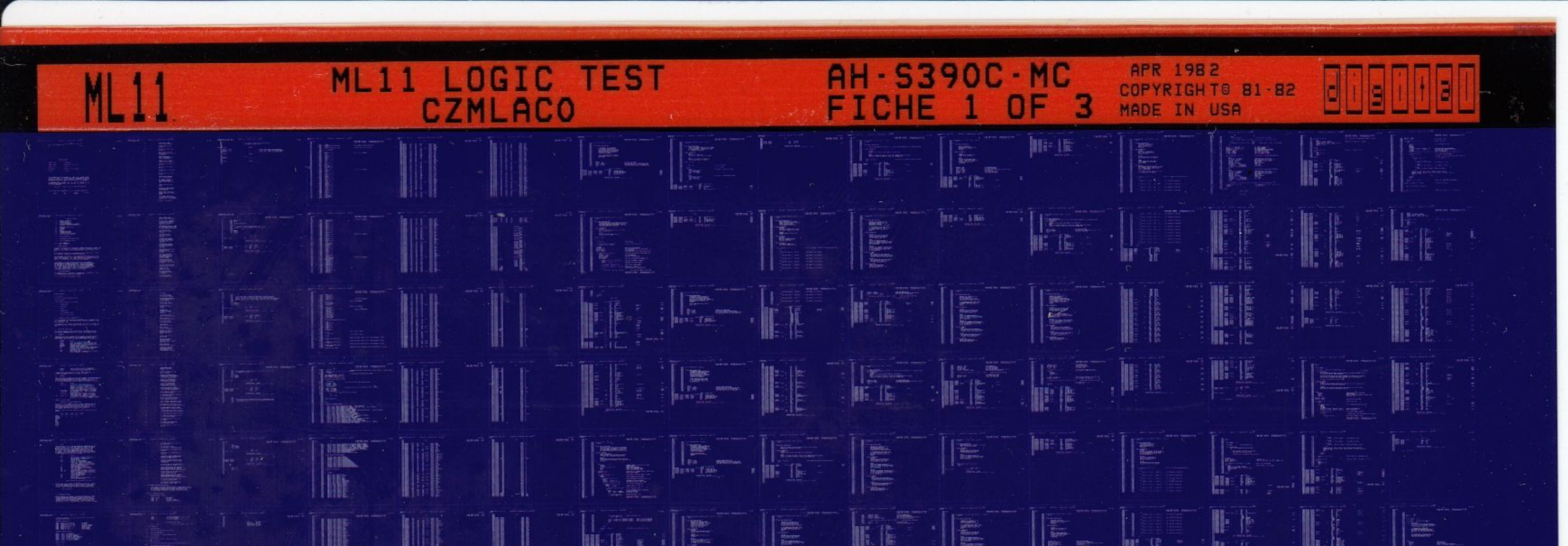
ML11

ML11 LOGIC TEST
CZMLACO

AH-S390C-MC
FICHE 1 OF 3

APR 1982
COPYRIGHT© 81-82
MADE IN USA



ML11

ML11 LOGIC TEST
CZMLACO

AH-S390C-MC
FICHE 2 OF 3

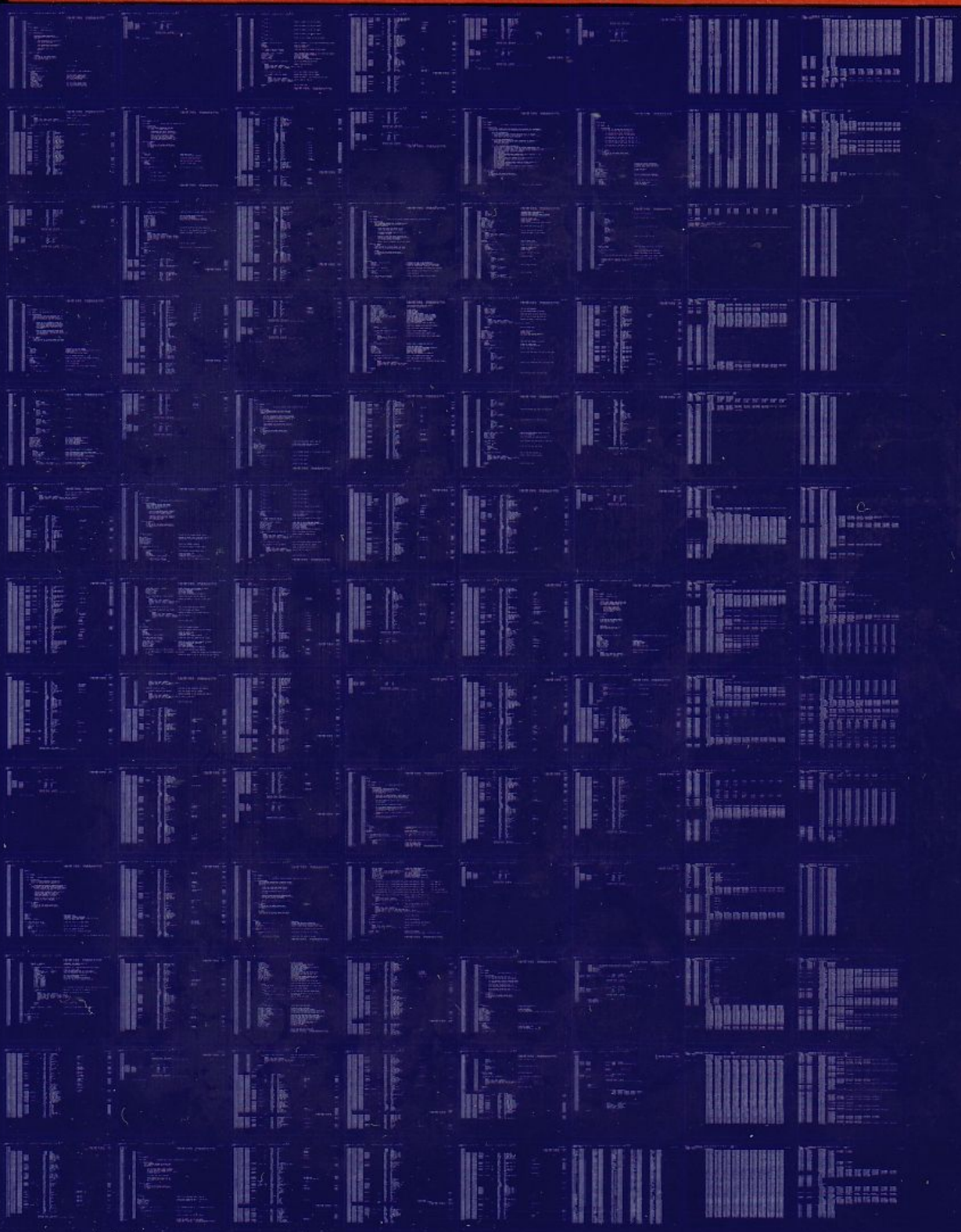
APR 1982
COPYRIGHT© 81-82
MADE IN USA

ML11

ML11 LOGIC TEST
CZMLACO

AH-S390C-MC
FICHE 3 OF 3

APR 1982
COPYRIGHT@ 81-82
MADE IN USA



```
      1                                 .TITLE CZMLACO ML-11 LOGIC TEST
                .SBTTL USER DOCUMENTATION
                                                                .REM &
```

IDENTIFICATION

PRODUCT CODE:	AC-S388C-MC
PRODUCT NAME:	CZMLACO ML11 LOGIC TEST
PRODUCT DATE:	19-MAY-81
MAINTAINER:	STORAGE SYSTEMS DIAGNOSTICS
AUTHOR:	D.W.NEALE

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

NO RESPONSIBILITY IS ASSUMED FOR THE USE OR RELIABILITY OF SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL OR ITS AFFILIATED COMPANIES.

COPYRIGHT (C) 1982 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL	PDP	UNIBUS	MASSBUS
DEC	DECUS	DECTAPE	

TABLE OF CONTENTS

1.0 GENERAL INFORMATION
1.1 PROGRAM ABSTRACT
1.2 SYSTEM REQUIREMENTS
1.3 RELATED DOCUMENTS AND STANDARDS
1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

2.0 OPERATING INSTRUCTIONS
2.1 COMMANDS
2.2 SWITCHES
2.3 FLAGS
2.4 HARDWARE QUESTIONS
2.5 SOFTWARE QUESTIONS
2.6 EXTENDED P-TABLE DIALOGUE
2.7 QUICK STARTUP PROCEDURE

3.0 ERROR INFORMATION

4.0 PERFORMANCE AND PROGRESS REPORTS

5.0 DEVICE INFORMATION TABLES

6.0 TEST SUMMARIES

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

DIAGNOSTIC ENGINEERING WAS CONTRACTED BY MEMORY ENGINEERING TO MAKE THIS DIAGNOSTIC IN ORDER TO AID MEMORY ENGINEERING TO DESIGN AND DEBUG THE ML-11, AID FIELD SERVICE FOR FIELD REPAIRS AND INSTALLATIONS OF THE ML-11.

THIS DIAGNOSTIC PRODUCT WILL BE DESIGNED TO TEST FROM ONE TO EIGHT ML-11 UNITS OFF A SINGLE RH11 OR RH70 CONTROLLER.

THE FUNCTIONAL LEVEL (FRU) OF THIS DIAGNOSTIC PRODUCT WILL BE TO THE LOGIC FUNCTION LEVEL (I.E. DRIVE SELECTION). UPON DETECTION OF AN ERROR BY THE DIAGNOSTIC, THE LOGIC FUNCTION AND RESPECTIVE MODULE WHICH IT IS LOCATED ON WILL BE PRINTED TO THE OPERATOR.

THIS DIAGNOSTIC HAS BEEN WRITTEN FOR USE WITH THE DIAGNOSTIC RUNTIME SERVICES SOFTWARE (SUPERVISOR). THESE SERVICES PROVIDE THE INTERFACE TO THE OPERATOR AND TO THE SOFTWARE ENVIRONMENT. THIS PROGRAM CAN BE USED WITH XXDP+, ACT, APT, SLIDE AND PAPER TAPE. FOR A COMPLETE DESCRIPTION OF THE RUNTIME SERVICES, REFER TO THE XXDP+ USER'S MANUAL. THERE IS A BRIEF DESCRIPTION OF THE RUNTIME SERVICES IN SECTION 2 OF THIS DOCUMENT.

1.2 SYSTEM REQUIREMENTS

THE HARDWARE DESIGN IS EXPECTED TO CONFORM TO THE STANDARDS SET FORTH IN THE MASSBUS SPECIFICATION (DEC STANDARD 159).

PDP-11 WITH MINIMUM OF 28K WORDS OF MEMORY

CONSOLE TERMINAL

RH11 OR RH70

1 TO 8 ML-11 DRIVES ON INTERMIXED BUS

XXDP+ LOAD MEDIA

1.3 RELATED DOCUMENTS AND STANDARDS

1. SUPPRGC.DOC
2. SUPINT.MEN
3. SUPFUN.C
4. XXDPPLUS.DOC
5. BLISS LANGUAGE GUIDE
6. BLISS-16 USER'S GUIDE

1.4 DIAGNOSTIC HIERARCY PREREQUISITES

IT WILL BE ASSUMED THAT PRIOR TO THE RUNNING OF THIS DIAGNOSTIC THAT ALL APPROPRIATE CPU, MAIN MEMORY AND RH CONTROLLER DIAGNOSTICS HAVE BEEN SUCCESSFULLY RUN.

THIS DIAGNOSTIC WILL HOWEVER PERFORM MINIMAL RH TESTS TO ENSURE ITS EXISTANCE AND BASIC FUNCTIONALITY BEFORE LOGIC TESTS ARE ALLOWED TO RUN.

2.0 OPERATING INSTRUCTIONS

THIS SECTION CONTAINS A BRIEF DESCRIPTION OF THE RUNTIME SERVICES. FOR DETAILED INFORMATION, REFER TO THE XXDP+ USER'S MANUAL (CHQUS).

2.1 COMMANDS

THERE ARE ELEVEN LEGAL COMMANDS FOR THE DIAGNOSTIC RUNTIME SERVICES (SUPERVISOR). THIS SECTION LISTS THE COMMANDS AND GIVES A VERY BRIEF DESCRIPTION OF THEM. THE XXDP+ USER'S MANUAL HAS MORE DETAILS.

COMMAND	EFFECT
START	START THE DIAGNOSTIC FROM AN INITIAL STATE
RESTART	START THE DIAGNOSTIC WITHOUT INITIALIZING
CONTINUE	CONTINUE AT TEST THAT WAS INTERRUPTED (AFTER ^C)
PROCEED	CONTINUE FROM AN ERROR HALT
EXIT	RETURN TO XXDP+ MONITOR (XXDP+ OPERATION ONLY!)
ADD	ACTIVATE A UNIT FOR TESTING (ALL UNITS ARE CONSIDERED TO BE ACTIVE AT START TIME
DROP	DEACTIVATE A UNIT
PRINT	PRINT STATISTICAL INFORMATION (IF IMPLEMENTED BY THE DIAGNOSTIC - SECTION 4.0)

DISPLAY	TYPE A LIST OF ALL DEVICE INFORMATION
FLAGS	TYPE THE STATE OF ALL FLAGS (SEE SECTION 2.3)
ZFLAGS	CLEAR ALL FLAGS (SEE SECTION 2.3)

A COMMAND CAN BE RECOGNIZED BY THE FIRST THREE CHARACTERS. SO YOU MAY, FOR EXAMPLE, TYPE "STA" INSTEAD OF "START".

2.2 SWITCHES

THERE ARE SEVERAL SWITCHES WHICH ARE USED TO MODIFY SUPERVISOR OPERATION. THESE SWITCHES ARE APPENDED TO THE LEGAL COMMANDS. ALL OF THE LEGAL SWITCHES ARE TABULATED BELOW WITH A BRIEF DESCRIPTION OF EACH. IN THE DESCRIPTIONS BELOW, A DECIMAL NUMBER IS DESIGNATED BY "DDDDD".

SWITCH	EFFECT
/TESTS:LIST	EXECUTE ONLY THOSE TESTS SPECIFIED IN THE LIST. LIST IS A STRING OF TEST NUMBERS, FOR EXAMPLE - /TESTS:1:5:7-10. THIS LIST WILL CAUSE TESTS 1,5,7,8,9,10 TO BE RUN. ALL OTHER TESTS WILL NOT BE RUN.
/PASS:DDDDD	EXECUTE DDDDD PASSES (DDDDD = 1 TO 64000)
/FLAGS:FLGS	SET SPECIFIED FLAGS. FLAGS ARE DESCRIBED IN SECTION 2.3.
/EOP:DDDDD	REPORT END OF PASS MESSAGE AFTER EVERY DDDDD PASSES ONLY. (DDDDD = 1 TO 64000)
/UNITS:LIST	TEST/ADD/DROP ONLY THOSE UNITS SPECIFIED IN THE LIST. LIST EXAMPLE - /UNITS:0:5:10-12 USE UNITS 0,5,10,11,12 (UNIT NUMBERS = 0-63)

EXAMPLE OF SWITCH USAGE:

```
START/TESTS:1-5/PASS:1000/EOP:100
```

THE EFFECT OF THIS COMMAND WILL BE: 1) TESTS 1 THROUGH 5 WILL BE EXECUTED, 2) ALL UNITS WILL TESTED 1000 TIMES AND 3) THE END OF PASS MESSAGES WILL BE PRINTED AFTER EACH 100 PASSES ONLY. A SWITCH CAN BE RECOGNIZED BY THE FIRST THREE CHARACTERS. YOU MAY, FOR EXAMPLE, TYPE "/TES:1-5" INSTEAD OF "/TESTS:1-5".

BELOW IS A TABLE THAT SPECIFIES WHICH SWITCHES CAN BE USED BY EACH COMMAND.

	TESTS	PASS	FLAGS	EOP	UNITS
START	X	X	X	X	X
RESTART	X	X	X	X	X
CONTINUE		X	X	X	
PROCEED			X		
DROP					X
ADD					X
PRINT					
DISPLAY					X
FLAGS					
ZFLAGS					
EXIT					

2.3 FLAGS

FLAGS ARE USED TO SET UP CERTAIN OPERATIONAL PARAMETERS SUCH AS LOOPING ON ERROR. ALL FLAGS ARE CLEARED AT STARTUP AND REMAIN CLEARED UNTIL EXPLICITLY SET USING THE FLAGS SWITCH. FLAGS ARE ALSO CLEARED AFTER A START COMMAND UNLESS SET USING THE FLAG SWITCH. THE ZFLAGS COMMAND MAY ALSO BE USED TO CLEAR ALL FLAGS. WITH THE EXCEPTION OF THE START AND ZFLAGS COMMANDS, NO COMMANDS AFFECT THE STATE OF THE FLAGS; THEY REMAIN SET OR CLEARED AS SPECIFIED BY THE LAST FLAG SWITCH.

FLAG	EFFECT
HOE	HALT ON ERROR - CONTROL IS RETURNED TO RUNTIME SERVICES COMMAND MODE
LOE	LOOP ON ERROR
IER*	INHIBIT ALL ERROR REPORTS
IBE*	INHIBIT ALL ERROR REPORTS EXCEPT FIRST LEVEL (FIRST LEVEL CONTAINS ERROR TYPE, NUMBER, PC, TEST AND UNIT)
IXE*	INHIBIT EXTENDED ERROR REPORTS (THOSE CALLED BY PRINTX MACRO'S)
PRI	DIRECT MESSAGES TO LINE PRINTER
PNT	PRINT TEST NUMBER AS TEST EXECUTES
BOE	"BELL" ON ERROR
UAM	UNATTENDED MODE (NO MANUAL INTERVENTION)
ISR	INHIBIT STATISTICAL REPORTS (DOES NOT APPLY TO DIAGNOSTICS WHICH DO NOT SUPPORT STATISTICAL REPORTING)
IDU	INHIBIT PROGRAM DROPPING OF UNITS
ADR	EXECUTE AUTODROP CODE
LOT	LOOP ON TEST
EVL	EXECUTE EVALUATION (ON DIAGNOSTICS WHICH HAVE EVALUATION SUPPORT)

*ERROR MESSAGES ARE DESCRIBED IN SECTION 3.1

SEE THE XXDP+ USER'S MANUAL FOR MORE DETAILS ON FLAGS. YOU MAY SPECIFY MORE THAN ONE FLAG WITH THE FLAG SWITCH. FOR EXAMPLE, TO CAUSE THE PROGRAM TO LOOP ON ERROR, INHIBIT ERROR REPORTS AND TYPE A "BELL" ON ERROR, YOU MAY USE THE FOLLOWING STRING:

```
/FLAGS:LOE:IER:BOE
```

2.4 HARDWARE QUESTIONS

WHEN A DIAGNOSTIC IS STARTED, THE RUNTIME SERVICES WILL PROMPT THE USER FOR HARDWARE INFORMATION BY TYPING "CHANGE HW (L) ?" YOU MUST ANSWER "Y" AFTER A START COMMAND UNLESS THE HARDWARE INFORMATION HAS BEEN "PRELOADED" USING THE SETUP UTILITY (SEE CHAPTER 6 OF THE XXDP+ USER'S MANUAL). WHEN YOU ANSWER THIS QUESTION WITH A "Y", THE RUNTIME SERVICES WILL ASK FOR THE NUMBER

```
OF UNITS (IN DECIMAL).  YOU WILL THEN BE ASKED THE FOLLOWING
QUESTIONS FOR EACH UNIT.

PARAMETER CODING CALLS

GPRMA   MSGH1,0,0,0,177777,YES          ;RH ADDRESS
GPRMD   MSGH2,2,0,77,11,70,YES          ;RH TYPE
GPRMD   MSGH3,4,0,777,0,777,YES         ;RH VECTOR ADDRESS
GPRMD   MSGH4,6,D,77,1,16.,YES          ;NUMBER FO ARRAYS
GPRML   MSGH5,10,1,YES                  ;DRIVE OPTIONS
GPRMD   MSGH6,12,0,7,0,7,YES            ;DRIVE NUMBER
GPRML   MSGH7,14,1,YES                  ;PARITY DISABLED

PARAMETER CODING MESSAGES

MSGH1:  .ASCIZ  /RH ADDRESS?/
MSGH2:  .ASCIZ  /IS RH AN '70' OR '11?/
MSGH3:  .ASCIZ  /RH VECTOR ADDRESS?/
MSGH4:  .ASCIZ  /NUMBER OF ARRAY MODULES?/
MSGH5:  .ASCIZ  /IS DRIVE OPTION AN ML11A?/
MSGH6:  .ASCIZ  /ML-11 DRIVE NUMBER?/
MSGH7:  .ASCIZ  /IS PARITY DISABLED?/

SAMPLE DIALOGE

DR> STA  <CR>

CHANGE HW <L>  ?  Y  <CR>

# UNITS <D>  ? 1  <CR>

UNIT 0
RH ADDRESS <O> 176400  ?  <CR>
IS RH AN '70' OR '11' <O>  ?  <CR>
RH VECTOR ADDRESS <O>  204  ?  <CR>
  MBER OF ARRAYS MODULES  ? <D>  16  ?  14  <CR>
  RIVE OPTION AN ML11A ? <L>  Y  ?  <CR>
ML-11 DRIVE NUMBER ? <O>  0  ?  <CR>
IS PARITY DISABLED ? <L>  N  ?  <CR>


2.5  SOFTWARE QUESTIONS

AFTER YOU HAVE ANSWERED THE HARDWARE QUESTIONS OR AFTER A RESTART
OR CONTINUE COMMAND, THE RUNTIME SERVICES WILL ASK FOR SOFTWARE
PARAMETERS.  THESE PARAMETERS WILL GOVERN SOME DIAGNOSTIC SPECIFIC
OPERATION MODES.  YOU WILL BE PROMPTED BY "CHANGE SW (L) ?"
IF YOU WISH TO CHANGE ANY PARAMETERS, ANSWER BY TYPING
"Y".  THE SOFTWARE QUESTIONS AND THE DEFAULT VALUES ARE DESCRIBED
IN THE NEXT PARAGRAPH(S).


PARAMETER CODING CALLS

GPRML  MSGS1,0,1,NO                     ;PRINT THE DRIVE SERIAL NUMBER
```

PARAMETER CODING MESSAGES

MSG1: .ASCIZ /PRINT SERIAL NO.?/

SAMPLE DIALOGE

PRINT SERIAL NO. ? <L> N ? Y <CR>

2.6 EXTENDED P-TABLE DIALOGUE

WHEN YOU ANSWER THE HARDWARE QUESTIONS, YOU ARE BUILDING ENTRIES IN A TABLE THAT DESCRIBES THE DEVICES UNDER TEST. THE SIMPLEST WAY TO BUILD THIS TABLE IS TO ANSWER ALL QUESTIONS FOR EACH UNIT TO BE TESTED. IF YOU HAVE A MULTIPLEXED DEVICE SUCH AS A MASS STORAGE CONTROLLER WITH SEVERAL DRIVES OR A COMMUNICATION DEVICE WITH SEVERAL LINES, THIS BECOMES TEDIOUS SINCE MOST OF THE ANSWERS ARE REPETITIOUS.

TO ILLUSTRATE A MORE EFFICIENT METHOD, SUPPOSE YOU ARE TESTING A FICTIONAL DEVICE, THE XY11. SUPPOSE THIS DEVICE CONSISTS OF A CONTROL MODULE WITH EIGHT UNITS (SUB-DEVICES) ATTACHED TO IT. THESE UNITS ARE DESCRIBED BY THE OCTAL NUMBERS 0 THROUGH 7. THERE IS ONE HARDWARE PARAMETER THAT CAN VARY AMONG UNITS CALLED THE Q-FACTOR. THIS Q-FACTOR MAY BE 0 OR 1. BELOW IS A SIMPLE WAY TO BUILD A TABLE FOR ONE XY11 WITH EIGHT UNITS.

```
# UNITS (D) ?  8<CR>

UNIT 1
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  0<CR>
Q-FACTOR (O) 0 ?  1<CR>

UNIT 2
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  1<CR>
Q-FACTOR (O) 1 ?  0<CR>

UNIT 3
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  2<CR>
Q-FACTOR (O) 0 ?  <CR>

UNIT 4
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  3<CR>
Q-FACTOR (O) 0 ?  <CR>

UNIT 5
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  4<CR>
Q-FACTOR (O) 0 ?  <CR>

UNIT 6
CSR ADDRESS (O) ?  160000<CR>
```

```
SUB-DEVICE # (0) ?  5<CR>
Q-FACTOR (0) 0 ?  <CR>

UNIT 7
CSR ADDRESS (0) ?  160000<CR>
SUB-DEVICE # (0) ?  6<CR>
Q-FACTOR (0) 0 ?  1<CR>

UNIT 8
CSR ADDRESS (0)  160000<CR>
SUB-DEVICE # (0) ?  7<CR>
Q-FACTOR (0) 1 ?  <CR>
```

NOTICE THAT THE DEFAULT VALUE FOR THE Q-FACTOR CHANGES WHEN A NON-DEFAULT RESPONSE IS GIVEN. BE CAREFUL WHEN SPECIFYING MULTIPLE UNITS!

AS YOU CAN SEE FROM THE ABOVE EXAMPLE, THE HARDWARE PARAMETERS DO NOT VARY SIGNIFICANTLY FROM UNIT TO UNIT. THE PROCEDURE SHOWN IS NOT VERY EFFICIENT.

THE RUNTIME SERVICES CAN TAKE MULTIPLE UNIT SPECIFICATIONS HOWEVER. LET'S BUILD THE SAME TABLE USING THE MULTIPLE SPECIFICATION FEATURE.

```
# UNITS (D) ?  8<CR>

UNIT 1
CSR ADDRESS (0) ?  160000<CR>
SUB-DEVICE # (0) ?  0,1<CR>
Q-FACTOR (0) 0 ?  1,0<CR>

UNIT 3
CSR ADDRESS (0) ?  160000<CR>
SUB-DEVICE # (0) ?  2-5<CR>
Q-FACTOR (0) 0 ?  0<CR>

UNIT 7
CSR ADDRESS (0) ?  160000<CR>
SUB-DEVICE # (0) ?  6,7<CR>
Q-FACTOR (0) 0 ?  1<CR>
```

AS YOU CAN SEE IN THE ABOVE DIALOGUE, THE RUNTIME SERVICES WILL BUILD AS MANY ENTRIES AS IT CAN WITH THE INFORMATION GIVEN IN ANY ONE PASS THROUGH THE QUESTIONS. IN THE FIRST PASS, TWO ENTRIES ARE BUILT SINCE TWO SUB-DEVICES AND Q-FACTORS WERE SPECIFIED. THE SERVICES ASSUME THAT THE CSR ADDRESS IS 160000 FOR BOTH SINCE IT WAS SPECIFIED ONLY ONCE. IN THE SECOND PASS, FOUR ENTRIES WERE BUILT. THIS IS BECAUSE FOUR SUB-DEVICES WERE SPECIFIED. THE "-" CONSTRUCT TELLS THE RUNTIME SERVICES TO INCREMENT THE DATA FROM THE FIRST NUMBER TO THE SECOND. IN THIS CASE, SUB-DEVICES 2, 3, 4 AND 5 WERE SPECIFIED. (IF THE SUB-DEVICE WERE SPECIFIED BY ADDRESSES, THE INCREMENT WOULD BE BY 2 SINCE ADDRESSES MUST BE ON AN EVEN BOUNDARY.) THE CSR ADDRESSES AND Q-FACTORS FOR THE FOUR ENTRIES ARE ASSUMED TO BE 160000 AND 0 RESPECTIVELY SINCE THEY WERE ONLY SPECIFIED ONCE. THE LAST TWO UNITS ARE SPECIFIED IN THE THIRD PASS.

THE WHOLE PROCESS COULD HAVE BEEN ACCOMPLISHED IN ONE PASS AS SHOWN BELOW.

```
# UNITS (D) ?  8<CR>

UNIT 1
CSR ADDRESS (O) ?  160000<CR>
SUB-DEVICE # (O) ?  0-7<CR>
Q-FACTOR (O) 0 ?  0,1,0,,,,1,1<CR>
```

AS YOU CAN SEE FROM THIS EXAMPLE, NULL REPLIES (COMMAS ENCLOSING A NULL FIELD) TELL THE RUNTIME SERVICES TO REPEAT THE LAST REPLY.

2.7 QUICK START-UP PROCEDURE (XXDP+)

TO START-UP THIS PROGRAM:

1. BOOT XXDP+
2. GIVE THE DATE AND ANSWER THE LSI AND 50HZ (IF THERE IS A CLOCK) QUESTIONS
3. TYPE "R NAME", WHERE NAME IS THE NAME OF THE BIN OR BIC FILE FOR THIS PROGRAM
4. TYPE "START"
5. ANSWER THE "CHANGE HW" QUESTION WITH "Y"
6. ANSWER ALL THE HARDWARE QUESTIONS
7. ANSWER THE "CHANGE SW" QUESTION WITH "N"

WHEN YOU FOLLOW THIS PROCEDURE YOU WILL BE USING ONLY THE DEFAULTS FOR FLAGS AND SOFTWARE PARAMETERS. THESE DEFAULTS ARE DESCRIBED IN SECTIONS 2.3 AND 2.5.

3.0 ERROR INFORMATION

3.1 TYPES OF ERROR MESSAGES

THERE ARE THREE LEVELS OF ERROR MESSAGES THAT MAY BE ISSUED BY A DIAGNOSTIC: GENERAL, BASIC AND EXTENDED. GENERAL ERROR MESSAGES ARE ALWAYS PRINTED UNLESS THE "IER" FLAG IS SET (SECTION 2.3). THE GENERAL ERROR MESSAGE IS OF THE FORM:

```
NAME  TYPE  NUMBER  ON UNIT NUMBER  TST NUMBER PC:XXXXXX
ERROR MESSAGE

,WHERE; NAME = DIAGNOSTIC NAME
        TYPE = ERROR TYPE (SYS FATAL, DEV FATAL, HARD OR SOFT)
        NUMBER = ERROR NUMBER
        UNIT NUMBER = 0 - N (N IS LAST UNIT IN PTABLE)
        TST NUMBER = TEST AND SUBTEST WHERE ERROR OCCURRED
        PC:XXXXXX = ADDRESS OF ERROR MESSAGE CALL
```

BASIC ERROR MESSAGES ARE MESSAGES THAT CONTAIN SOME ADDITIONAL INFORMATION ABOUT THE ERROR. THESE ARE ALWAYS PRINTED UNLESS THE "IER" OR "IBR" FLAGS ARE SET (SECTION 2.3). THESE MESSAGES ARE PRINTED AFTER THE ASSOCIATED GENERAL MESSAGE.

EXTENDED ERROR MESSAGES CONTAIN SUPPLEMENTARY ERROR INFORMATION SUCH AS REGISTER CONTENTS OR GOOD/BAD DATA. THESE ARE ALWAYS PRINTED UNLESS THE "IER", "IBR" OR "IXR" FLAGS ARE SET (SECTION 2.3). THESE MESSAGES ARE PRINTED AFTER THE ASSOCIATED GENERAL ERROR MESSAGE AND ANY ASSOCIATED BASIC ERROR MESSAGES.

3.2 SPECIFIC ERROR MESSAGES

ERROR MESSAGES ARE HANDLED VIA A DICTIONARY STRUCTURE.

WORDS AND PHRASES ARE MULTIPLY REFERENCED USING ONLY ONE COPY OF THE WORD OR PHRASE IN CORE.

THIS PERMITS EXTENSIVE ERROR MESSAGE PRINTING AT MINIMAL STORAGE REQUIREMENTS.

THE FOLLOWING DEMONSTRATES TYPICAL ERROR MESSAGES:

```
ML11 DVC FTL ERR 00077 ON UNIT 07 TST 027 SUB 002 PC: 050432
ASYNCHRONOUS MODULE FAILURE
EXCESSIVE DATA ERRORS DURING INITIAL ARR RD_WRT

ML11 DVC FTL ERR  00112 ON UNIT 03 TST 037 SUB 000 PC: 056466
ASYNCHRONOUS MODULE FAILURE
ARRAY ADRS MULTIPLEXER FAILURE
FAILED AT DSA: 000000
```

3.2.1 ERROR NUMBER DEFINITION

ERROR NO.	FAILING LOGIC
1	DRIVE DID NOT RESPOND WITHIN 1.5 US
2	DSA REG READ/WRITE ERROR DURING DRIVE SEL TEST
3	UNIQUE DRIVE SELECTION ERROR
4	ML REGISTER READ WRITE ONES/ZEROES ERROR
5	ML REGISTER READ WRITE ONES/ZEROES ERROR
6	ML REGISTER INITIALIZATION ERROR
7	CONTROL BUS BAD PARITY

NOT DETECTED

8 CONTROL BUS GOOD PARITY NOT DETECTED.

9 CONTROL BUS BAD PARITY GENERATED.

10 ARRAY SIZING LOGIC ERROR

11 GO BIT NOT CLR AFTER NOOP FUNCTION

12 ILF BIT SET DURING NOOP FUNCTION

13 OPI BIT SET DURING NOOP FUNCTION

14 GO BIT NOT SET DURING WRITE CHECK FUNCTION

15 DRY BIT NOT CLEAR DURING WRITE CHECK FUNCTION

16 DRY BIT SET WHEN GO SET DURING WRITE CHECK FUNCTION

17 ILF SET DURING WRITE CHECK FUNCTION

18 OPI BIT SET DURING WRITE CHECK FUNCTION

19 GO BIT NOT CLEAR AFTER WRITE CHECK FUNCTION 'ASYNC FAILURE'

20 GO BIT NOT CLEAR AFTER WRITE CHECK FUNCTION 'SYNC FUNCTION'

21 DRY BIT NOT SET AFTER WRITE CHECK FUNCTION

22 GO BIT NOT CLEAR AFTER WRITE CHECK FUNCTION 'ASYNC FAILURE'

23 GO BIT NOT CLEAR AFTER WRITE CHECK FUNCTION 'SYNC FAILURE'

24 GO BIT NOT SET DURING WRITE FUNCTION

25 DRY BIT CLEAR WITH GO CLEAR DURING WRITE FUNCTION.

26 DRY BIT SET WITH GO BIT DURING WRITE FUNCTION.

```
                    27          ILF BIT SET DURING WRITE
                                FUNCTION

                    28          OPI BIT SET DURING WRITE FUNCTION

                    29          GO BIT NOT CLEAR AFTER WRITE
                                FUNCTION 'ASYNC FAILURE'

                    30          GO BIT NOT CLEAR AFTER WRITE
                                FUNCTION 'SYNC FAILURE'

                    31          DRY BIT NOT SET AFTER WRITE
                                FUNCTION.

                    32          GO BIT NOT CLEAR AFTER WRITE
                                FUNCTION 'ASYNC FAILURE'

                    33          GO BIT NOT CLEAR AFTER WRITE
                                FUNCTION 'ASYNC FAILURE'

                    34          GO BIT NOT SET DURING READ
                                FUNCTION.

                    35          DRY BIT SET WHILE GO CLEAR
                                DURING READ FUNCTION.

                    36          DRY BIT SET WHILE GO SET
                                DURING READ FUNCTION.

                    37          ILF BIT SET DURING READ
                                FUNCTION.

                    38          OPI BIT SET DURING READ
                                FUNCTION

                    39          GO BIT NOT CLEAR AFTER READ
                                FUNCTION 'ASYNC FAILURE'

                    40          GO BIT NOT CLEAR AFTER READ
                                FUNCTION 'SYNC FAILURE'

                    41          DRY BIT NOT SET AFTER READ
                                FUNCTION

                    42          GO BIT NOT CLEAR AFTER READ
                                FUNCTION 'ASYNC FAILURE'

                    43          GO BIT NOT CLEAR AFTER READ
                                FUNCTION 'SYNC FAILURE'

                    44          GO BIT NOT CLEAR AFTER
                                CLEAR FUNCTION

                    45          DRY BIT SET WHILE GO SET
                                DURING CLEAR FUNCTION
```

46	DRY BIT NOT SET AFTER CLEAR FUNCTION.
47	ILF BIT SET DURING CLEAR FUNCTION
48	OPI BIT SET DURING CLEAR FUNCTION
50	COMPOSITE ERROR BIT NOT SET AFTER MLER BITS SET
51	ATA BIT SETTING ERROR
52	ATTN BIT SETTING ERROR
53	ATTN BIT NOT CLEARED BY MLCS1 NOOP FUNCTION
54	ATA BIT NOT CLEAR AFTER WRITING A ONE TO IT
55	WRITING A ONE TO OTHER DRIVES ATA BIT CLEARED THIS DRIVES ATA BIT
56	GO BIT NOT CLEARED AFTER SEARCH FUNCTION
57	ILF BIT SET DURING SEARCH FUNCTION
58	NO-OP FUNCTION DID NOT CLEAR THE ATA BIT AFTER BEING SET
59	ATA BIT NOT SET AFTER SEARCH FUNCTION
60	OPI BIT SET DURING SEARCH FUNCTION AT PRESENT ARRAYS
61	ATA BIT SET AFTER SEARCH AT NOT PRESENT ARRAYS.
62	GO BIT NOT CLEAR AFTER READ-IN-PRESET FUNCTION
63	ILF BIT SET DURING READ-IN-SET FUNCTION
64	OPI BIT SET DURING READ-IN-PRESET FUNCTION
65	UV BIT NOT SET AFTER READ-IN-PRESET

66	GO BIT NOT CLEARED AFTER ILLEGAL FUNCTION
67	ILLEGAL FUNCTION NOT DETECTED
68	OPI BIT SET WITH ILLEGAL FUNCTION
69	RMR BIT NOT SET AFTER MODIFYING REG WITH FUNCTION IN PROGRESS.
70	MEMORY ARRAY PROM CHECK SUM ERRORS DURING INITIAL PROM READS
71	NIBBLE OFF SET COUNTS GREATER THAN 14 DETECTED.
72	UNS BIT SET WITH GOOD UV DATA
73	UNS BIT SET WITH GOOD UV DATA
74	UNS BIT NOT SET WITH BAD UV DATA
75	UNS BIT NOT SET WITH BAD UV DATA
76	MEMORY ARRAY PROM ROW/ COL DATA ORING ERROR
77	BAD NIBBLE THRESHOLD OF 36 EXCEEDED DURING INITIAL ARRAY READ/WRITE TEST
78	UNIQUE PROM SELECTION ERROR
79	FAILURE TO FIND GOOD ROW DURING READ WRITE ARRAY WITH PROM DATA
80	MEMORY ARRAY TIMING AND CONTROL FAILURE TO REFRESH MEMORY
81	DATA ERRORS DETECTED AT LAST BLOCK DURING ADDRESSES COUNTER TEST. (TEST ABORTED)
82	ADDRESS COUNTER ERROR

83	UNIQUE MEMORY ARRAY MODULE SELECTION FAILURE
84	ALL BITS IN ALL NIBBLES TESTED DURING SEQUENCER EXISTENCE TEST WERE IN ERROR (FAIL UNIT)
85	INTERMEDIATE FAILURE. SOME BITS IN NIBBLES TESTED WERE IN ERROR (CONTINUE TESTING)
86	SYNC BUS DATA BIT WRITE PATH CONTINUITY FAILURE
87	SYNC BUS DATA BIT READ PATH CONTINUITY FAILURE.
88	RAM BUS ADRS COUNTER FAILURE TO LOAD/UNLOAD SKIP RAM DURING WRITE FUNCTION
89	RAM BUS ADRS COUNTER FAILURE TO LOAD/UNLOAD SKIP RAM DURING READ FUNCTION.
90	SYNC DATA BUS WRITE PATH UNIQUE DATA BIT FAILURE (ALL ONES NIBBLE PATTERN)
91	SYNC DATA BUS WRITE PATH UNIQUE DATA BIT FAILURE (SHIFTED BIT NIBBLE PATTERN)
92	SYNC DATA BUS WRITE PATH UNIQUE DATA BIT FAILURE (ALL ONES NIBBLE PATTERN)
93	SYNC DATA BUS READ PATH UNIQUE DATA BIT FAILURE
94	NIBBLE OFF SET COUNTERS FAILURE
95	CS1 FUNCTION ABORT FAILURE DURING CLASS 'A' ERROR.
96	CS1 FUNCTION ABORT FAILURE DURING CLASS

'B' ERROR

97	LBT BIT SET BEFORE A LAST BLOCK TRANSFER
98	DSA REGISTER INCREMENT FAILURE DURING NON LAST BLOCK TRANSFERS.
99	LBT BIT NOT CLEAR AFTER LOADING DSA REG
100	LBT BIT NOT SET AFTER A LAST BLOCK TRANSFER
101	DSA REGISTER INCREMENT FAILURE AFTER A LAST BLOCK TRANSFER
102	IAE BIT NOT SET AT INVALID SECTOR ADDRESSES
103	AOE BIT NOT SET AFTER ADDRESS OVERFLOW
104	SC BIT NOT AFTER CS1 FUNCTION ADORT
105	GOOD DATA BUS PARITY NOT DETECTED
106	GOOD DATA BUS PARITY NOT GENERATED
107	UNS BIT SET AFTER WRITING TO A SECTOR DURING PROM DATA TEST
108	UNS BIT NOT SET WITH BAD UV DATA
109	WCE BIT SET DURING MBUS WRITE/READ FUNCTION TROUBLE SHOOTING LOOP TEST
110	UNIQUE REGISTER SELECTION TEST FAILURE
111	FAILURE TO FIND GOOD MOS RAM ROW DURING ARRAY ADRS MUX TEST (INTERMEDIATE DIAG MSG)

112	UNIQUE ARRAY MODULE ROW/COL ADDRESSING FAILURE
113	DRIVE TYPE REGISTER VALUE WAS NOT CORRECT
114	TRE BIT SET UNEXPECTEDLY DURING A WRITE CHECK TRANSFER (INTERMEDIATE DIAG ERROR)
115	TRE BIT SET UNEXPECTEDLY DURING A WRITE TRANSFER (INTERMEDIATE DIAG ERROR)
116	TRE BIT SET UNEXPECTEDLY DURING A READ TRANSFER (INTERMEDIATE DIAG ERROR)
117	TRE BIT DID NOT SET AFTER A REGISTER MODIFICATION ERROR (EXCEPTION WAS NOT ASSERTED)
118	DATA DIAGNOSTIC REGISTER D1 D2 E2 INITIALIZATION ERRORS
119	RH CONTROLLER FAILURE TO RESPOND TO EXISTANCE PROBE
120	NED BIT SET DURING MASS BUS TO UNIBUS COMMUNICATION PROBE
121	DATA DIAGNOSTIC REGISTER D1 D2 E2 ONE'S / ZEROE'S READ WRITE ERRORS
122	DATA DIAGNOSTIC REGISTER D1 D2 E2 SHIFTING 1'S AND 0'S READ WRITE ERRORS
123	ECC HARD ERROR BIT NOT SET WHEN UNCORRECTABLE ECC ERRORS WERE READ
124	ECC ERROR REGISTER FAILURE TO LATCH FAILING ERROR CORRECTION INFORMATION
125	ECC ERROR REGISTER FAILURE TO CLEAR
126	ECC ERROR LOCATION REGISTER FAILED TO CLOCK IN DSA ADDRESS
127	ECC ERROR LOCATION REGISTER DATA BIT ERRORS
128	ECC ERROR LOCATION REGISTER INITIALIZATION ERRORS
129	ECC HARD ERROR BIT FAILED TO SET NOT SET

130 DATA CHECK ERROR BIT FAILED TO SET / NOT SET

131 CRC DATA BUS FAILURE. BIT CONTINUITY / UNIQUENESS ERRORS

132 UNIQUE NIBBLE CRC GENERATION CODE FAILURES

133 UNIQUE WORD CRC GENERATION CODE FAILURES

134 CORRECTABLE ERROR SYNDRONE DECODE FAILURE. FAILURE TO COMPLIMENT EXPECTED FAILING BIT(S)

135 CORRECTABLE ERROR SYNDRONE DECODE FAILURE. UNEXPECTED BIT(S) FOUND COMPLIMENTED

136 ECH OR UNC BIT NOT SET DURING READS WITH ECC ERRORS IN CHANNEL > 35

137 ERROR CORRECTION WAS NOT INHIBITED DURING UNCORRECTABLE ECC ERRORS. UNEXPECTED BIT(S) WERE FOUND COMPLIMENTED

138 ECH OR UNC BIT NOT SET WHEN MULTIPLE CHANNEL ERRORS WERE READ

139 ERROR CORRECTION WAS NOT INHIBITED DURING UNCORRECTABLE ECC ERROR. UNEXPECTED BIT(S) WERE FOUND COMPLIMENTED

140 ECC FAILURE TO DETECT AND CORRECT SINGLE BIT CHANNEL ERRORS

141 ECC FAILURE TO DETECT AND CORRECT MULTIPLE BIT CHANNEL ERRORS

4.0 PERFORMANCE AND PROGRESS REPORTS

AT THE END OF EACH PASS, THE PASS COUNT IS GIVEN ALONG WITH THE TOTAL NUMBER OF ERRORS REPORTED SINCE THE DIAGNOSTIC WAS STARTED. THE "EOP" SWITCH CAN BE USED TO CONTROL HOW OFTEN THE END OF PASS MESSAGE IS PRINTED. SECTION 2.2 DESCRIBES SWITCHES.

5.0 DEVICE INFORMATION TABLES

HARDWARE DEFAULT PTABLE

```
.WORD   176400          ;RH ADDRESS
.WORD   70              ;RH TYPE
.WORD   204             ;RH VECTOR ADDRESS
.WORD   16.             ;NUMBER OF ARRAY MODULES
```

```
                .WORD   1               ;IS DRVE OPTION ML11A,  1=16K,  0=64K
                .WORD   0               ;ML-11 DRIVE NUMBER
                .WORD   0               ;IS PARITY DISABLED,  1=YES,  0=NO

                                        SOFTWARE DEFAULT TABLE

                PRSN:   .WORD   1               ;PRINT SERIAL NUMBER, 1=YES,  0=NO
```

6.0 TEST SUMMARIES

TST1. MASS BUS READY

TEST THE RH CONTROLLER FOR EXISTANCE

TST2. MASS BUS HANDSHAKE

TEST MASS BUS ---> UNIBUS COMMUNICATIONS

TST3. DRIVE PRESENT

TEST TO SEE IF THE DRIVE UNDER TEST EXIST.

TST4. DRIVE SELECTION

SEE IF SELECTING OTHER DRIVES ON RH EFFECTS DRIVE UNDER TEST.

TST5. READ WRITE REGISTER ONE'S ZERO'S TEST

TEST REGISTERS READ WRITE CAPABILITY AND UNIQUENESS

TST6. READ WRITE REGISTER SHIFTING ONE'S AND ZEROES

TEST REGISTERS FOR UNIQUE DATA BITS.

TST7. REGISTER INITIALIZATION

TEST REGISTERS FOR CORRECT INIT DATA.

TST8 REGISTER SELECTION TEST

TEST FOR UNIQUE REGISTER SELECTION

TST9. PRINT DRIVE SERIAL NUMBER

PRINT THE CONTENTS OF MLSN IF THE SOFTWARE QUESTION WAS ANSWERED 'YES'.

TST10. C-BUS PARITY

TEST IF DRIVE CAN DETECT BAD PARITY ON C-BUS AND GENERATE GOOD PARITY.

TST11. MEMORY SIZING

SEE IF MEMORY SIZING LOGIC DETECTS AND RECORD CORRECT NUMBER OF ARRAYS PRESENT.

TST12. NO-OP FUNCTION

SEE IF A NO-OP FUNCTION CAN BE EXECUTED.

TST13. WRITE CHECK FUNCTION

SEE IF A WRITE CHECK FUNCTION CAUSES THE DRIVE TO HANG.

TST14. WRITE FUNCTION

SEE IF A WRITE FUNCTION CAUSES THE DRIVE TO HANG.

TST15. READ FUNCTION

SEE IF A READ FUNCTION CAUSES THE DRIVE TO HANG.

TST16. CLEAR FUNCTION

SEE IF A CLEAR FUNCTION CAN BE EXECUTED.

TST17. DIAGNOSTIC REGISTER READ WRITE

TEST THE DIAGNOSTIC REGISTERS FOR
1'S/0'S , SHIFTING 1'S/0'S AND
INITIALIZATION

TST18. COMPOSITE ERROR BIT TEST

SEE IF EACH INDIVIDUAL ERROR BIT IN MLER CAUSES A COMPOSITE ERROR.

TST19. ATA BIT

TEST IF THE ATA BIT CAN BE SET AND CLEARED.

TST20. SEARCH FUNCTION

SEE IF A SEARCH FUNCTION CAN BE EXECUTED ON ALL PRESENT ARRAYS.

TST21. READ IN PRESET

TEST IF A READ IN PRESET FUNCTION SETS VOL V H.

TST22. ILLEGAL FUNCTION

SEE IF WRITING AN ILLEGAL FUNCTION TO CS1 CAN BE DETECTED AND THAT A TRANSFER IS NOT INITIATED.

TST23. REGISTER MODIFICATION REFUSED

TEST TO SEE IF WRITING TO SPECIFIC REGISTERS ARE ABORTED WHILE THE DRIVE IS ACTIVE. SEE IF WRITING TO NON-SPECIFJC REGISTERS ARE ALLOWED WHILE DRIVE IS ACTIVE.

TST24. INITIAL PROM TEST

TEST PROMS FOR EXISTENCE.

TST25. PROM 'OR' FUNCTION TEST

TEST THE PROM DATA ORING FUNC

TST26. UV ERROR TEST

TEST ABILITY OF UV ERR PROMS TO DETECT ALL POSSIBLE CHECK SUM ERRORS.

TST27. INITIAL ARRAY TEST

TEST ARRAY TIMING AND CONTROLFOR EXISTENCE.

TST28. PROM SELECTION TEST

TEST FOR UNIQUE PROM SELECTION.

TST29. READ WRITE MEMORY ARRAY WITH PROM DATA (DIAG MODE)

SEE IF MEMORY CAN BE WRITTEN AND READ.

ALSO FIND ERROR FREE BLOCK OF MEMORY FOR FUTURE TESTS.

TST30. REFRESH TIMING

TEST TO SEE IF MEMORY CAN BE REFRESHED.

TST31. ADDRESS COUNTER

TEST THE ADDRESS COUNTER FOR ABILITY TO COUNT THROUGH ALL POSSIBLE MEMORY ADDRESSES.

TST32. ARRAY MODULE SELECTION

TEST FOR UNIQUE ARRAY MODULE SELECTION

TST33. SEQUENCER EXISTENCE TEST

TEST TO SEE IF BASIC SEQUENCER TIMING EXISTS.

TST34. SYNC DATA BUS CONTINUITY/WRITE PATH

TEST SYNCHRONOUS DATA BUS WRITE PATH FOR CONTINUITY BY READ-ING WRITING ONE'S AND ZERO'S.

TST35. SYNC DATA BUS CONTINUITY/READ PATH

TEST SYNCHRONOUS DATA BUS READ PATH FOR CONTINUITY BY READING WRITING ONE'S AND ZEROES.

TST36. RAM-BUS ADDRESS COUNTER/WRITE PATH

TEST ABILITY OF THE RAM-BUS ADDRESS COUNTERS TO LOAD/UNLOAD

THE SKIP DURING WRITE FUNCTIONS.

TST37. RAM BUS ADRS COUNTER/READ PATH

TEST ABILITY OF RAM/BUS ADRS COUNTERS TO LOAD/UNLOAD THE SKIP RAM DURING READ FUNCTIONS.

TST38. SYNC DATA BUS BIT UNIQUENESS/WRITE PATH

TEST SYNCHRONOUS DATA BUS FOR DATA BIT UNIQUENESS BY WRITING SHIFTING PATTERNS OF ONE'S AND ZERO'S TO THE ML.

TST39. SYNC DAT BUS BIT UNIQUENESS/READ PATH

TEST SYNCHRONOUS DATA BUS READ PATH FOR DATA BIT UNIQUENESS BY WRITING SHIFTING PATTERNS OF ONES AND ZEROES TO THE

TST40. ARRAY ADDRESS MUX

TEST FOR UNIQUE ROW AND COLUMN ADDRESSING

TST41. NIBBLE OFFSET

TEST NIBBLE OFFSET COUNTERS TO COUNT TO 14 NIBBLE DATA TO BE SHIFTED ON DETECTION OF BAD NIBBLES.

TST42. CS1 FUNCTION ABORT

SEE IF A CLASS 'B' ERROR ABORTS A FUNCTION WHILE IN PROGRESS.

SEE IF A CLASS'A' ERROR IS DETECTED BUT FUNCTION IS ALLOWED TO COMPLETE.

TST43. LAST BLOCK INDICATOR

TEST THE LAST BLOCK INDICATOR BIT FOR NOT SETTING BELOW THE LAST AND SETTING AND CLRING AT THE LAST BLOCK

TST44. INVALID ADDRESS TEST

FOR ALL ILLEGAL DSA ADDRESSES READ THE IAE BIT SET.

TST45. ADDRESS OVERFLOW

TEST FOR AOE ON TRANSFERS WHICH EXTEND BEYOND THE LAST BLOCK.

TST46. SYNC BUS PARITY

TEST FOR BAD PARITY DETECTION AND GOOD PARITY GENERATION.

TST47. WRITE READ MEMORY ARRAY (M-BUS BLOCK MODE)

WRITE READ MEMORY VIA M-BUS BLOCK WITH MINIMUM OVERHEAD

TST48. TEST THE CRC DATA BUS BETWEEN THE CRC GENERATORS AND THE CRC/MBUS DATA

MUX FOR CONTINUITY AND BIT UNIQUENESS

TST49. TEST CRC CODES GENERATED FOR ONE CRC GROUP (52 UNIQUE NIBBLES)

TST50. TEST CRC CODES GENERATED FOR ONE CRC GROUP (13 UNIBUS WORDS)

TST51. TEST SYNDRONE DECODE AND ERROR CORRECTION TO DECODE AND CORRECT SINGLE BIT AND MULTIPLE BIT CHANNEL ERRORS

TST52. TEST SYNDRONE DECODE TO DETECT BUT NOT CORRECT UNCORRECTABLE CHANNEL ERRORS

TST53. TEST SYNDRONE DECODE TO DETECT BUT NOT CORRECT UNCORRECTABLE MULTIPLE CHANNEL ERRORS

TST54. TEST SYNDRONE GENERATION, SYNDRONE DECODE AND ERROR CORRECTION FOR SIGLE BIT CHANNEL ERRORS

TST55. TEST SYNDRONE GENERATION, SYNDRONE DECODE AND ERROR CORRECTION FOR MULTIPLE BIT CHANNEL ERRORS

TST56. TEST THE ECC ERROR REGISTER FOR CLEARING AND LATCHING OF ECC ERROR INFORMATION ON DETECTION OF ECC ERRORS

TST57. TEST THE ECC ERROR LOCATION REGISTER FOR CLOCKING, BIT UNIQUENESS, CLEARING AND LATCHING

TST58. VIA ECC_DM AND ECC_DIS TEST THE ECH BIT FOR SETTING AND NOT SETTING

TST59. VIA ECC_EN, ECC_DM AND ECC_DIS TEST THE DCK BIT FOR SETTING AND NOT SETTING

TST60. FROM DATA TEST

VERIFY THAT CHECK SUM VALUES FOR ALL PROM LOCATIONS ARE CORRECT.

&

```
                        .SBTTL  PROGRAM HEADER AND TABLES

                        .ENABL  ABS,AMA
002000                  .       =       2000

002000                  BGNMOD

                        ;++
                        ; THE PROGRAM HEADER IS THE INTERFACE BETWEEN
                        ; THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
                        ;--

002000                  POINTER ALL


002000                  HEADER  CZMLA,C,0,1800.,0

                        ;
                        ;       NAMES OF DEVICES SUPPORTED BY THIS PROGRAM
                        ;
002122                  DEVTYP <ML-11>



                        ;
                        ;       TEST DESCRIPTION
                        ;
002130                  DESCRIPT <ML-11 LOGIC TEST>


                        ;
                        ;       THE GLOBAL ERROR TABLE (INFORMATION
                        ;       USED IN A CALL TO THE MACRO 'ERROR')
                        ;

002152                  ERRTBL
002152  000000          ERRTYP::        .WORD   0
002154  000000          ERRNBR::        .WORD   0
002156  000000          ERRMSG::        .WORD   0
002160  000000          ERRBLK::        .WORD   0

                        ;++
                        ; THE DISPATCH TABLE CONTAINS THE STARTING ADDRES OF EACH TEST.
                        ; IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
                        ;--

002162                  DISPATCH 60


                        ;++
                        ; THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
                        ; THE TEST-DEVICE PARAMETERS.  THE STRUCTURE OF THIS TABLE
                        ; IS IDENTICAL TO THE STRUCTURE OF THE HARDWARE P-TABLES,
                        ; AND IS USED AS A "TEMPLATE" FOR BUILDING THE P-TABLES.
                        ;--
```

```
                                        BGNHW   DFPTBL
                                        .WORD   176400          ;RH ADDRESS
                                        .WORD   70              ;RH TYPE
                                        .WORD   204             ;RH VECTOR ADDRESS
                                        .WORD   16.             ;NUMBER OF ARRAY MODULES
                                        .WORD   1               ;IS DRVE OPTION ML11A,  1=16K,  0=64K
                                        .WORD   0               ;ML-11 DRIVE NUMBER
                                        .WORD   0               ;IS PARITY DISABLED,  1=YES,  0=NO

                                        ENDHW



                                ;++
                                ; THE DEFAULT SOFTWARE P-TABLE CONTAINS VARIOUS DATA USED BY THE
                                ; PROGRAM AS OPERATIONAL PARAMETERS.  THESE PARAMETERS ARE SET
                                ; UP AT ASSEMBLY TIME AND MAY BE VARIED BY THE OPERATOR AT RUN
                                ; TIME.
                                ;--

                                        BGNSW   SFPTBL

                                PRSN:   .WORD   1               ;PRINT SERIAL NUMBER, 1=YES,  0=NO

                                        ENDSW
```

```
                                        ;++
                                        ; THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
                                        ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
                                        ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
                                        ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
                                        ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
                                        ; WITH THE OPERATOR.
                                        ;--

002400                                          BGNHRD

002402                                          GPRMA   MSGH1,0,0,0,177777,YES
002412                                          GPRMD   MSGH2,2,0,77,11,70,YES
002424                                          GPRMD   MSGH3,4,0,777,0,777,YES
002436                                          GPRMD   MSGH4,6,D,77,1,16.,YES
002450                                          GPRML   MSGH5,10,1,YES
002456                                          GPRMD   MSGH6,12,0,7,0,7,YES
002470                                          GPRML   MSGH7,14,1,YES

002476                                          ENDHRD

002476  122  110  040  MSGH1:  .ASCIZ  /RH ADDRESS?/
002512  111  123  040  MSGH2:  .ASCIZ  /IS RH AN '70' OR '11?/
002540  122  110  040  MSGH3:  .ASCIZ  /RH VECTOR ADDRESS?/
002563  116  125  115  MSGH4:  .ASCIZ  /NUMBER OF ARRAY MODULES?/
002614  111  123  040  MSGH5:  .ASCIZ  /IS DRIVE OPTION AN ML11A?/
002646  115  114  055  MSGH6:  .ASCIZ  /ML-11 DRIVE NUMBER?/
002672  111  123  040  MSGH7:  .ASCIZ  /IS PARITY DISABLED?/
                                .EVEN

                                        ;++
                                        ; THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS
                                        ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
                                        ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
                                        ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
                                        ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
                                        ; WITH THE OPERATOR.
                                        ;--

002716                                          BGNSFT

002720                                          GPRML   MSGS1,0,1,YES
                                                .EVEN

002726                                          ENDSFT

002726  120  122  111  MSGS1:  .ASCIZ  /PRINT SERIAL NO.?/              ;PRINT DRIVE SERIAL NUMBER?
                                .EVEN
```

```
                                ;++
                                ; THIS TABLE IS USED BY THE RUNTIME SERVICES
                                ; TO PROTECT THE LOAD MEDIA.
                                ;--

002750                                  BGNPROT

002750  177777                         -1              ;OFFSET INTO P-TABLE FOR CSR ADDRESS
002752  177777                         -1              ;OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
002754  177777                         -1              ;OFFSET INTO P-TABLE FOR DRIVE NUMBER

002756                                  ENDPROT


002756                          $PATCH::
002756                                  .BLKW   20

003016                                  ENDMOD
```

```
      6 ;ML3                                                    1-May-1981 10:49:46    TOPS-20 Bliss-16 V2(212)
      7 ;                                                       1-May-1981 10:49:37    PA:<NEALE>ML3.BLI.1 (1)
      8
      9 ;          0001  MODULE ML3 =
     10 ;          0002  BEGIN
     11 ;          0003
     12 ;          0004  REQUIRE 'BLSMAC.REQ';
     13 ;          1494
     14 ;          1495  !+
     15 ;          1496  !  THE REPORT CODING SECTION CONTAINS THE
     16 ;          1497  ! ''PRINTS'' CALLS THAT GENERATE STATISTICAL REPORTS.
     17 ;          1498  !-
     18 ;          1499
     19 ;          1500  BGNRPT;
     20 ;          1501  RETURN;
     21 ;          1502  ENDRPT;
     25
     26
     27                                        .SBTTL  LRPT
     28
     32 004152  000207                 LRPT:   RTS     PC                              ;                        1493
     33
     34                                 ; Routine Size:  1 word
     35                                 ; Maximum stack depth per invocation:  0 words
     40
     41
     45
     46                                        .SBTTL  L$RPT
     50 004154  004767  177772          L$RPT:: JSR     PC,LRPT                         ;                        1501
     51 004160  104425                          TRAP    25
     52 004162  000207                          RTS     PC
     53
     54                                 ; Routine Size:  4 words
     55                                 ; Maximum stack depth per invocation:  0 words
```

```
                        1503
                        1504    !+
                        1505    ! THE AUTODROP CODE IS EXECUTED IMMEDIATELY AFTER THE INITIALIZE
                        1506    ! CODE IF THE "ADR" FLAG WAS SET.  THE UNIT(S) UNDER TEST ARE
                        1507    ! CHECKED TO SEE IF THEY WILL RESPOND.  THOSE THAT DON'T ARE IMMEDIATELY
                        1508    ! DROPPED FROM TESTING.  ISSUE A "DODU" FOR THOSE THAT DON'T RESPOND.
                        1509    !-
                        1510
                        1511    BGNAUTO;
                        1512    RETURN;
                        1513    ENDAUTO;
                                                .SBTTL  LAUTO
004164  000207                  LAUTO:  RTS     PC                              ;       1502
                                ; Routine Size:  1 word
                                ; Maximum stack depth per invocation:  0 words
                                                .SBTTL  L$AUTO
004166  004767  177772          L$AUTO::JSR     PC,LAUTO                        ;       1512
004172  104461                          TRAP    61
004174  000207                          RTS     PC
                                ; Routine Size:  4 words
                                ; Maximum stack depth per invocation:  0 words
                        1514    ;
```

```
                                                                    1-May-1981 10:49:46    TOPS-20 Bliss-16 V2(212)
                                                                    1-May-1981 10:49:37    PA:<NEALE>ML3.BLI.1 (1)
                                                                                                                    ;ML3
                                                                                                                    :
                                                                                                                    
                1515  !+                                                                                            ;
                1516  ! THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE                             ;
                1517  ! TO NO LONGER BE TESTED.                                                                     ;
                1518  !-                                                                                            ;
                1519                                                                                                ;
                1520  BGNDU;                                                                                        ;
                1521  RETURN;                                                                                       ;
                1522  ENDDU;                                                                                        ;
                                                                                                                    
                                                                                                                    
                                        .SBTTL  LDU
004176  000207                  LDU:    RTS     PC                                      ;                          1513
                                                                                                                    
                                        ; Routine Size:  1 word
                                        ; Maximum stack depth per invocation:  0 words
                                                                                                                    
                                                                                                                    
                                                                                                                    
                                        .SBTTL  L$DU
004200  004767  177772          L$DU::  JSR     PC,LDU                                  ;                          1521
004204  104453                          TRAP    53
004206  000207                          RTS     PC
                                                                                                                    
                                        ; Routine Size:  4 words
                                        ; Maximum stack depth per invocation:  0 words
                                                                                                                    
                1523                                                                                                ;
                1524  !+                                                                                            ;
                1525  ! THE ADD-UNIT SECTION CONTAINS ANY CODE THE PROGRAMMER WISHES                               ;
                1526  ! TO BE EXECUTED IN CONJUNCTION WITH THE ADDING OF A UNIT BACK                                ;
                1527  ! TO THE TEST CYCLE.                                                                          ;
                1528  !-                                                                                            ;
```

```
    172 ;ML3                                                    1-May-1981 10:49:46    TOPS-20 Bliss-16 V2(212)
    173 ;                                                       1-May-1981 10:49:37    PA:<NEALE>ML3.BLI.1 (1)
    174
    175 ;          1529
    176 ;          1530  BGNAU;
    177 ;          1531  RETURN;
    178 ;          1532  ENDAU;
    182
    183                                    .SBTTL  LAU
    187 004210  000207                LAU: RTS     PC                                ;                          1522
    188
    189                                ; Routine Size:  1 word
    190                                ; Maximum stack depth per invocation:  0 words
    195
    196
    200
    201                                    .SBTTL  L$AU
    205 004212  004767  177772        L$AU:: JSR     PC,LAU                          ;                          1531
    206 004216  104452                     TRAP    52
    207 004220  000207                     RTS     PC
    208
    209                                ; Routine Size:  4 words
    210                                ; Maximum stack depth per invocation:  0 words
    215
    216
    217 ;          1533  END
    218 ;          1534
    219 ;          1535  ELUDOM
    223
    224
    225
    226
```

```
                        ;ML3                                                    1-May-1981 10:49:46    TOPS
                        ;                                                       1-May-1981 10:49:37    PA:<
                        ; Size:          20 code + 0 data words
                        ; Run Time:      00:03.6
                        ; Elapsed Time: 00:12.2
                        ; Memory Used:  24 pages
                        ; Compilation Complete
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (1)

                0001  MODULE ML4 =
                0002  BEGIN
                0003  !
                0004  ! PRETTY BLF COMMANDS
                0005  !
                0006  !<BLF/NOERROR>
                0007  !<BLF/LOWERCASE_KEY>
                0008  !
                0009  ! REQUIRE
                0010  !
                0011
                0012  require 'BLSMAC.REQ';                    !BLISS INTERFACE MODULE
                1502
                1503  !
                1504  ! CONSTANT LITERALS
                1505  !
                1506
                1507  literal
                1508      ONE = 1,                            !DATA BIT OF ONE
                1509      ONES = %o'177777',                  !DATA PATTERN OF ONES
                1510      ZERO = 0,                           !DATA BIT OF ZERO
                1511      ZEROES = 0,                         !DATA PATTERN OF ZEROES
                1512      NUM_OF_REG = 22,                    !NUMBER OF BLOCKS IN GLOBAL STORAGE 'ML-REG'
                1513      FIELD_SIZ = 4,                      !FIELD SIZ FOR FIELD DECLARATIONS 'WORD_MAP'
                1514      !
                1515      !MLCS1 FUNCTION CODES
                1516      !
                1517      NOOP = 1,                           !NOOP FUNCTION
                1518      DRV_CLR = %o'11',                   !DRIVE CLEAR FUNCTION
                1519      RD_IN_PRE = %o'21',                 !READ IN PRESET FUNCTION
                1520      SEARCH = %o'31',                    !SEARCH FUNCTION
                1521      WRT_CHK = %o'51',                   !WRITE CHECK FUNCTION
                1522      write = %o'61',                     !WRITE FUNCTION
                1523      read = %o'71',                      !READ FUNCTION
                1524      !
                1525      ! DELAY ARGUMENTS
                1526      !
                1527      ONE_US = 1,                         !ONE MICRO SECOND DELAY
                1528      FRTY_US = 40,                       !FORTY MICRO SECOND DELAY
                1529      TWO_TH_US = 2000;                   !TWO THOUSAND MICRO SECOND DELAY
                1530
                1531  !
                1532  ! FIELD DECLARATIONS
                1533  !
                1534
                1535  field
                1536      WORD_MAP =                          !MAPS GLOBAL STORAGE 'ML_REG' INTO REGISTER PERSONALITIES
                1537          set
                1538          REGISTER_ADD = [0, 0, 16, 0],   !REGISTERS ADDRESS
                1539          FORCE_HI = [1, 0, 16, 0],       !REGISTERS  FORCED HI BITS
                1540          FORCE_LO = [2, 0, 16, 0],       !REGISTERS FORCED LO BITS
                1541          DONT_CARE = [3, 0, 16, 0]       !REGISTERS IGNORE BITS
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (1)

                1542            tes,
                1543        NIB_MAP =                   !MAPS OWN STORAGE NIB_SAVE INTO TEN FOUR BIT NIBBLES
                1544            set
                1545            NIB_0 = [0, 0, 4, 0],   !NIBBLE 0 BITS <0:3>
                1546            NIB_1 = [0, 4, 4, 0],   !NIBBLE 1 BITS <4:7>
                1547            NIB_2 = [0, 8, 4, 0],   !NIBBLE 2 BITS <8:11>
                1548            NIB_3 = [0, 12, 4, 0],  !NIBBLE 3 BITS <12:15>
                1549            NIB_4 = [1, 0, 4, 0],   !NIBBLE 4 BITS <16:19>
                1550            NIB_5 = [1, 4, 4, 0],   !NIBBLE 5 BITS <20:23>
                1551            NIB_6 = [1, 8, 4, 0],   !NIBBLE 6 BITS <24:27>
                1552            NIB_7 = [1, 12, 4, 0],  !NIBBLE 7 BITS <28:31>
                1553            NIB_8 = [2, 8, 4, 0],   !NIBBLE 8 BITS <32:35>
                1554            NIB_9 = [2, 12, 3, 0]   !NIBBLE 9 BITS <36:39>
                1555            tes,
                1556        DT1_FLD =
                1557            set
                1558            E0_5 = [0, 0, 6, 0],
                1559            C0_5 = [0, 6, 6, 0],
                1560            CRC_DATA = [0, 13, 1, 0],
                1561            SGL_DATA = [0, 14, 1, 0],
                1562            UNC_DATA = [0, 15, 1, 0],
                1563            EE_DATA = [0, 0, 16, 0],
                1564            A0_5 = [1, 0, 6, 0],
                1565            P0_5 = [1, 6, 6, 0],
                1566            ECR_DATA = [1, 12, 1, 0],
                1567            B0_5 = [2, 0, 6, 0]
                1568            tes;
                1569
                1570    !<BLF/PAGE>
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (2)

;          1571  !
;          1572  ! OWN STORAGE
;          1573  !
;          1574
;          1575  own
;          1576      NIB_SAVE : block [3] field (NIB_MAP) volatile,
;          1577                                          !STORAGE LOCATION TO SAVE NIBBLE DATA READ DURING DIAG MODE
;          1578      HW_OR_TBL : vector [127] volatile,  !STORES HARDWARE ORED PROM DATA DURING PROM OR FUNC TEST
;          1579      PTBL_PTR : volatile,                !HARDWARE P-TABLE POINTER
;          1580      OP_NUM_ARR : volatile,              !OPERATORS NUMBER  OF ARRAY INPUTTED
;          1581      ARR_INC : volatile,                 !ARRAY SELECTION INCREMENT VALUE
;          1582      GOOD_BLK : volatile,                !GOOD BLOCK ADRS
;          1583      PAR_DIS : volatile,                 !PARITY DISABLE FLAG
;          1584      CHIP_SIZ : volatile,                !MOS RAM CHIP SIZE
;          1585      LST_BLK : volatile,                 !LAST ADDRESSABLE BLOCK
;          1586      ARR_16 : volatile,                  !MAX NUMBER OF ARRAY ALLOWED
;          1587      LST_ARR : volatile,                 !LAST ADDRESSABLE ARRAY
;          1588      IO_BUF : vector [256] volatile,     !INPUT OUTPUT BUFFER
;          1589      STK_OFF : vector [10, byte] volatile,  !STACK OFFSET STORAGE LOCATION
;          1590      stack : vector [198, byte] volatile,   !STACK OF 198 BYTE LOCATIONS
;          1591      PD_TEMP : bitvector [16] volatile,  !PROM DATA STORAGE LOCATION DURING DIAG MODES
;          1592      W_C_SIZE : volatile,                !STORES WORD COUNT FOR 16K OR 64K XFERS
;          1593      RAS_INC : volatile,                 !ROW ADRS STROBE INCREMENT
;          1594      WT_DATA : volatile,                 !SAVE WRITE DATA DURING REG READ WRITE TEST
;          1595      RD_DATA : volatile,                 !SAVE READ DATA DURING REG READ WRITE TEST
;          1596      DRIVE_TYPE : volatile,              !DRIVE TYPE STORAGE LOCATION
;          1597      LST_DOT_REG : volatile,             !LAST ML-11 REG INDEX FOR THIS TYPE RH CONTROLLER
;          1598      REG_INIT_FLG : initial (0) volatile,   !FLAG TO DETECT DOING REG INIT TEST
;          1599      A_CAL : volatile,                   !CALCULATED CRC STORAGE LOCATION
;          1600      B_CAL : volatile,                   !CALCULATED CRC STORAGE LOCATION
;          1601      P_CAL : volatile,                   !CALCULATED CRC STORAGE LOCATION
;          1602      A_GEN : volatile,                   !GENERATED CRC STORAGE LOCATION
;          1603      B_GEN : volatile,                   !GENERATED CRC STORAGE LOCATION
;          1604      P_GEN : volatile,                   !GENERATED CRC STORAGE LOCATION
;          1605  !
;          1606  !<BLF/NOFORMAT>
;          1607  !
;          1608     ML_REG: blockvector [NUM_OF_REG,FIELD_SIZ] field(WORD_MAP) !ML11 REGISTERS
;          1609        preset (                                                !STRUCTURE TO STORE INDIVIDULA
;          1610                                                                !REGISTER PERSONALITIES
;          1611                                                                !ie. REGISTER ADDRESSES
;          1612                                                                !BITS FORCED HIGH
;          1613                                                                !BITS FORCED LOW
;          1614                                                                !READ ONLY BITS
;          1615           [0,FORCE_HI]    = %o'004000',    !MLCS1
;          1616
;          1617           [0,FORCE_LO]    = %o'173701',
;          1618           [0,DONT_CARE]   = %o'160200',
;          1619
;          1620           [5,FORCE_LO]    = %o'25077',     !MLDS
;          1621           [5,FORCE_HI]    = %o'010600',
;          1622           [5,DONT_CARE]   = %o'000100',
```

```
151 ;ML4
152 ;
153
154 ;          1623
155 ;          1624          [6,FORCE_LO]    = %o'014620',   !MLER
156 ;          1625
157 ;          1626          [7,DONT_CARE]   = %o'177400',   !MLAS
158 ;          1627
159 ;          1628          [8,FORCE_LO]    = %o'100000',   !MLPA
160 ;          1629
161 ;          1630          [10,FORCE_LO]   = %o'000020',   !MLMR
162 ;          1631          [10,DONT_CARE]  = %o'177400',
163 ;          1632
164 ;          1633          [11,FORCE_HI]   = %o'000110',   !MLDT
165 ;          1634          [11,FORCE_LO]   = %o'177666',
166 ;          1635          [11,DONT_CARE]  = %o'000001',
167 ;          1636
168 ;          1637          [13,FORCE_LO]   = %o'140300',   !MLE1
169 ;          1638
170 ;          1639          [14,FORCE_LO]   = %o'100300',   !MLE2
171 ;          1640
172 ;          1641          [17,FORCE_LO]   = %o'010000',   !MLEE
173 ;          1642          [21,DONT_CARE]  = %o'000000'    !MLCS2
174 ;          1643                  ) volatile,
175 ;          1644
176 ;          1645
177 ;          1646  REM_TBL:vector [63,byte]          !REMAINDER TABLE
178 ;          1647    preset (                        !STRUCTURE TO STORE PRECALCULATED
179 ;          1648          [0] = %b'000001',         !CRC REMAINDER VALUES FOR CRC CODE
180 ;          1649          [1] = %b'000010',         !GENERATION
181 ;          1650          [2] = %b'000100',
182 ;          1651          [3] = %b'001000',
183 ;          1652          [4] = %b'010000',
184 ;          1653          [5] = %b'100000',
185 ;          1654          [6] = %b'000011',
186 ;          1655          [7] = %b'000110',
187 ;          1656          [8] = %b'001100',
188 ;          1657          [9] = %b'011000',
189 ;          1658          [10] = %b'110000',
190 ;          1659          [11] = %b'100011',
191 ;          1660          [12] = %b'000101',
192 ;          1661          [13] = %b'001010',
193 ;          1662          [14] = %b'010100',
194 ;          1663          [15] = %b'101000',
195 ;          1664          [16] = %b'010011',
196 ;          1665          [17] = %b'100110',
197 ;          1666          [18] = %b'001111',
198 ;          1667          [19] = %b'011110',
199 ;          1668          [20] = %b'111100',
200 ;          1669          [21] = %b'111011',
201 ;          1670          [22] = %b'110101',
202 ;          1671          [23] = %b'101001',
203 ;          1672          [24] = %b'010001',
204 ;          1673          [25] = %b'100010',
205 ;          1674          [26] = %b'000111',
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (2)

;          1675          [27] = %b'001110',
;          1676          [28] = %b'011100',
;          1677          [29] = %b'111000',
;          1678          [30] = %b'110011',
;          1679          [31] = %b'100101',
;          1680          [32] = %b'001001',
;          1681          [33] = %b'010010',
;          1682          [34] = %b'100100',
;          1683          [35] = %b'001011',
;          1684          [36] = %b'010110',
;          1685          [37] = %b'101100',
;          1686          [38] = %b'011011',
;          1687          [39] = %b'110110',
;          1688          [40] = %b'101111',
;          1689          [41] = %b'011101',
;          1690          [42] = %b'111010',
;          1691          [43] = %b'110111',
;          1692          [44] = %b'101101',
;          1693          [45] = %b'011001',
;          1694          [46] = %b'110010',
;          1695          [47] = %b'100111',
;          1696          [48] = %b'001101',
;          1697          [49] = %b'011010',
;          1698          [50] = %b'110100',
;          1699          [51] = %b'101011',
;          1700          [52] = %b'010101',
;          1701          [53] = %b'101010',
;          1702          [54] = %b'010111',
;          1703          [55] = %b'101110',
;          1704          [56] = %b'011111',
;          1705          [57] = %b'111110',
;          1706          [58] = %b'111111',
;          1707          [59] = %b'111101',
;          1708          [60] = %b'111001',
;          1709          [61] = %b'110001',
;          1710          [62] = %b'100001'}volatile,
;          1711
;          1712    DT_1:blockvector[5,3] field (DT1_FLD)          !DATA TABLE 1
;          1713    preset (                                       !STRUCTURE TO STORE TEST DATA
;          1714          [0,E0_5] = %b'111111',                   !FOR TEST 56 'ECC ERROR LOCATION TEST'
;          1715          [0,C0_5] = %b'100100',
;          1716          [0,CRC_DATA] = %b'1',
;          1717          [0,SGL_DATA] = %b'1',
;          1718          [0,UNC_DATA] = %b'0',
;          1719          [0,A0_5] = %b'000000',
;          1720          [0,P0_5] = %b'111111',
;          1721          [0,ECR_DATA] = %b'0',
;          1722          [0,B0_5] = %b'000000',
;          1723          [1,E0_5] = %b'111111',
;          1724          [1,C0_5] = %b'100100',
;          1725          [1,CRC_DATA] = %b'1',
;          1726          [1,SGL_DATA] = %b'1',
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (2)

;          1727          [1,UNC_DATA] = %b'0',
;          1728          [1,A0_5] = %b'111111',
;          1729          [1,P0_5] = %b'000000',
;          1730          [1,ECA_DATA] = %b'0',
;          1731          [1,B0_5] = %b'000000',
;          1732          [2,E0_5] = %b'111111',
;          1733          [2,C0_5] = %b'000101',
;          1734          [2,CRC_DATA] = %b'0',
;          1735          [2,SGL_DATA] = %b'0',
;          1736          [2,UNC_DATA] = %b'1',
;          1737          [2,A0_5] = %b'000000',
;          1738          [2,P0_5] = %b'111111',
;          1739          [2,ECA_DATA] = %b'1',
;          1740          [2,B0_5] = %b'111111',
;          1741          [3,E0_5] = %b'111111',
;          1742          [3,C0_5] = %b'000101',
;          1743          [3,CRC_DATA] = %b'0',
;          1744          [3,SGL_DATA] = %b'0',
;          1745          [3,UNC_DATA] = %b'1',
;          1746          [3,A0_5] = %b'000000',
;          1747          [3,P0_5] = %b'000000',
;          1748          [3,ECA_DATA] = %b'0',
;          1749          [3,B0_5] = %b'111111',
;          1750          [4,E0_5] = %b'111111',
;          1751          [4,C0_5] = %b'000101',
;          1752          [4,CRC_DATA] = %b'0',
;          1753          [4,SGL_DATA] = %b'0',
;          1754          [4,UNC_DATA] = %b'1',
;          1755          [4,A0_5] = %b'111111',
;          1756          [4,P0_5] = %b'000000',
;          1757          [4,ECA_DATA] = %b'1',
;          1758          [4,B0_5] = %b'111111')volatile,
;          1759
;          1760  !
;          1761  !<BLF/FORMAT>
;          1762  !
;          1763      RH_ADD : volatile,                  !RH CONTROLLER BASE ADDRESS
;          1764      RH_TYP : volatile,                  !RH CONTROLLER TYPE
;          1765      RH_VEC : volatile,                  !RH CONTROLLER VECTOR ADDRESS
;          1766      ML_LUN : volatile,                  !ML LOGICAL UNIT NO.
;          1767      ML_DUT : volatile;                  !ML DRIVE NUMBER
;          1768
;          1769  !
;          1770  !<BLF/PAGE>
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (3)

;          1771  !
;          1772  EQUALS;
;          1773  !
;          1774  ! MACRO DEFINITIONS
;          1775  !
;          1776
;          1777  macro
;          1778  !
;          1779  ! REGISTER NAMES:
;          1780  !
;        M 1781      MLCS1 =
;          1782   .ML_REG [0,REGISTER_ADD]%,          !CONTROL AND STATUS REGISTER 1
;        M 1783      MLWC =
;          1784   .ML_REG [1,REGISTER_ADD]%,          !WORD COUNT REGISTER
;        M 1785      MLBA =
;          1786   .ML_REG [2,REGISTER_ADD]%,          !UNIBUS ADDRESS REGISTER
;        M 1787      MLDA =
;          1788   .ML_REG [3,REGISTER_ADD]%,          !DESIRED ADDRESS REGISTER
;        M 1789      MLCS2 =
;          1790   .ML_REG [4,REGISTER_ADD]%,          !CONTROL AND STATUS REGISTER 2
;        M 1791      MLDS =
;          1792   .ML_REG [5,REGISTER_ADD]%,          !DRIVE STATUS REGISTER
;        M 1793      MLER =
;          1794   .ML_REG [6,REGISTER_ADD]%,          !ERROR REGISTER
;        M 1795      MLAS =
;          1796   .ML_REG [7,REGISTER_ADD]%,          !ATTENTION SUMMARY REGISTER
;        M 1797      MLLA =
;          1798   .ML_REG [8,REGISTER_ADD]%,          !LOOK AHEAD REGISTER
;        M 1799      MLPA =
;          1800   .ML_REG [8,REGISTER_ADD]%,          !PROM ADDRESS REGISTER
;        M 1801      MLDB =
;          1802   .ML_REG [9,REGISTER_ADD]%,          !DATA BUFFER REGISTER
;        M 1803      MLMR =
;          1804   .ML_REG [10,REGISTER_ADD]%,         !MAINTENANCE REGISTER
;        M 1805      MLDT =
;          1806   .ML_REG [11,REGISTER_ADD]%,         !DRIVE TYPE REGISTER
;        M 1807      MLSN =
;          1808   .ML_REG [12,REGISTER_ADD]%,         !SERIAL NUMBER REGISTER
;        M 1809      MLE1 =
;          1810   .ML_REG [13,REGISTER_ADD]%,         !ECC CRC WORD REGISTER 1
;        M 1811      MLE2 =
;          1812   .ML_REG [14,REGISTER_ADD]%,         !ECC CRC WORD REGISTER 2
;        M 1813      MLD1 =
;          1814   .ML_REG [15,REGISTER_ADD]%,         !DATA DIAGNOSTIC REGISTER 1
;        M 1815      MLD2 =
;          1816   .ML_REG [16,REGISTER_ADD]%,         'DATA DIAGNOSTIC REGISTER 2
;        M 1817      MLEE =
;          1818   .ML_REG [17,REGISTER_ADD]%,         !ECC ERROR REGISTER
;        M 1819      MLEL =
;          1820   .ML_REG [18,REGISTER_ADD]%,         !ECC ERROR LCOATION REGISTER
;        M 1821      MLPD =
;          1822   .ML_REG [19,REGISTER_ADD]%,         !PROM DATA REGISTER
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (3)

;        M 1823      MLBAE =
;          1824    .ML_REG [20,REGISTER_ADD]%,          !BUS ADDRESS EXTENSION REGISTER
;        M 1825      MLCS3 =
;          1826    .ML_REG [21,REGISTER_ADD]%,          !CONTROL AND STATUS REGISTER 3
;          1827    !
;          1828    ! BIT ASSIGNMENTS:
;          1829    !
;        M 1830      SC =
;          1831     (MLCS1)<15,1>%,                     !MLCS1 BIT ASSIGNMENTS
;        M 1832      TRE =
;          1833     (MLCS1)<14,1>%,
;        M 1834      MCPE =
;          1835     (MLCS1)<13,1>%,
;        M 1836      DVA =
;          1837     (MLCS1)<11,1>%,
;        M 1838      RDY =
;          1839     (MLCS1)<7,1>%,
;        M 1840      IE =
;          1841     (MLCS1)<6,1>%,
;        M 1842      GO =
;          1843    (MLCS1)<0,1>%,
;        M 1844      ML_FUNC =
;          1845     (MLCS1)<0,6>%,
;        M 1846      DLT =
;          1847     (MLCS2)<15,1>%,                     !MLCS2 BIT ASSIGNMENTS
;        M 1848      WCE =
;          1849     (MLCS2)<14,1>%,
;        M 1850      PE =
;          1851     (MLCS2)<13,1>%,
;        M 1852      NED =
;          1853     (MLCS2)<12,1>%,
;        M 1854      NEM =
;          1855     (MLCS2)<11,1>%,
;        M 1856      PGE =
;          1857     (MLCS2)<10,1>%,
;        M 1858      MXF =
;          1859     (MLCS2)<9,1>%,
;        M 1860      MDPE =
;          1861     (MLCS2)<8,1>%,
;        M 1862      ORDY =
;          1863     (MLCS2)<7,1>%,
;        M 1864      IRDY =
;          1865     (MLCS2)<6,1>%,
;        M 1866      CLR =
;          1867     (MLCS2)<5,1>%,
;        M 1868      PAT =
;          1869     (MLCS2)<4,1>%,
;        M 1870      BAI =
;          1871     (MLCS2)<3,1>%,
;        M 1872      DRV_NUM =
;          1873    (MLCS2)<0,3>%,
;        M 1874      ATTN =
```

```
423 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
424 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (3)
425
426 ;          1875     (MLDS)<15,1>%,                     !MLDS BIT ASSIGNMENTS
427 ;        M 1876       COMP_ERR =
428 ;          1877     (MLDS)<14,1>%,
429 ;        M 1878       MOL =
430 ;          1879     (MLDS)<12,1>%,
431 ;        M 1880       LBT =
432 ;          1881     (MLDS)<10,1>%,
433 ;        M 1882       DPR =
434 ;          1883     (MLDS)<8,1>%,
435 ;        M 1884       DRY =
436 ;          1885     (MLDS)<7,1>%,
437 ;        M 1886       VV =
438 ;          1887     (MLDS)<6,1>%,
439 ;        M 1888       DCK =
440 ;          1889     (MLER)<15,1>%,                     !MLER BIT ASSIGNMENTS
441 ;        M 1890       UNS =
442 ;          1891     (MLER)<14,1>%,
443 ;        M 1892       OPI =
444 ;          1893     (MLER)<13,1>%,
445 ;        M 1894       IAE =
446 ;          1895     (MLER)<10,1>%,
447 ;        M 1896       AOE =
448 ;          1897     (MLER)<9,1>%,
449 ;        M 1898       ECH_ERR =
450 ;          1899     (MLER)<6,1>%,
451 ;        M 1900       DPAR =
452 ;          1901     (MLER)<5,1>%,
453 ;        M 1902       CPAR =
454 ;          1903     (MLER)<3,1>%,
455 ;        M 1904       RMR =
456 ;          1905     (MLER)<2,1>%,
457 ;        M 1906       ILR =
458 ;          1907     (MLER)<1,1>%,
459 ;        M 1908       ILF =
460 ;          1909     (MLER)<0,1>%,
461 ;        M 1910       ARR_TYP =
462 ;          1911     (MLMR)<10,1>%,                     !MLMR BIT ASSIGNMENTS
463 ;        M 1912       ML_NUM_ARR =
464 ;          1913     (MLMR)<11,5>%,
465 ;        M 1914       REF_MAR =
466 ;          1915     (MLMR)<7,1>%,
467 ;        M 1916       PROM_RW =
468 ;          1917     (MLMR)<6,1>%,
469 ;        M 1918       PROM_DIS =
470 ;          1919     (MLMR)<5,1>%,
471 ;        M 1920       DAT_CLK =
472 ;          1921     (MLMR)<4,1>%,
473 ;        M 1922       DAT_DM =
474 ;          1923     (MLMR)<3,1>%,
475 ;        M 1924       DCK_EN =
476 ;          1925     (MLMR)<2,1>%,
477 ;        M 1926       ECC_DIS =
```

```
479 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
480 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (3)
481 ;
482 ;          1927   (MLMR)<1,1>%,
483 ;        M 1928      ECC_DM =
484 ;          1929    (MLMR)<0,1>%,
485 ;        M 1930      DRV_TYP =
486 ;          1931    (MLDT)<0,1>%,                          !MLDT BIT ASSIGNMENTS
487 ;        M 1932      CRC_A =
488 ;        M 1933                                  !MLE1 BIT ASSIGNMENTS
489 ;          1934   (MLE1)<0,6>%,
490 ;        M 1935      PAR_CRC_WRD =
491 ;          1936   (MLE1)<8,6>%,
492 ;        M 1937      CRC_B =
493 ;        M 1938                                  !MLE2 BIT ASSIGNMENTS
494 ;          1939   (MLE2)<0,6>%,
495 ;        M 1940      UNC_ERR =
496 ;          1941    (MLEE)<15,1>%,                         !MLEE BIT ASSIGNMENTS
497 ;        M 1942      SGL_ERR =
498 ;          1943    (MLEE)<14,1>%,
499 ;        M 1944      CRC_ERR =
500 ;          1945    (MLEE)<13,1>%,
501 ;        M 1946      BIT_IN_ERR =
502 ;          1947   (MLEE)<0,6>%,
503 ;        M 1948      CHAN_IN_ERR =
504 ;          1949   (MLEE)<6,6>%,
505 ;          1950   !
506 ;          1951   ! MISCELLANEOUS MACRO DEFINITIONS:
507 ;          1952   !
508 ;        M 1953      SN3 =
509 ;          1954   (MLSN)<12,4>%,
510 ;        M 1955      SN2 =
511 ;          1956   (MLSN)<8,4>%,
512 ;        M 1957      SN1 =
513 ;          1958   (MLSN)<4,4>%,
514 ;        M 1959      SN0 =
515 ;          1960   (MLSN)<0,4>%,
516 ;        M 1961      IS_SET =
517 ;        M 1962                                  !TEST IF BIT IS EQUAL 1
518 ;          1963   eql 1%,
519 ;        M 1964      IS_NOT_SET =
520 ;        M 1965                                  !TEST IF BIT IS EQUAL 0
521 ;          1966   eql 0%,
522 ;        M 1967      HI =
523 ;        M 1968                                  !READS REGISTERS FORCED HI BITS FROM PERSONALITY TABLE
524 ;          1969   ML_REG[.index,FORCE_HI]%,
525 ;        M 1970      LO =
526 ;        M 1971                                  !READS REGISTERS FORCED LO BITS FROM PERSONALITY TABLE
527 ;          1972   ML_REG[.index,FORCE_LO]%,
528 ;        M 1973      IGNORE =
529 ;        M 1974                                  !READS REGISTERS DONT_CARE BITS FROM PERSONALITY TABLE
530 ;          1975   ML_REG[.index,DONT_CARE]%,
531 ;        M 1976      MLE2_MASK =
532 ;        M 1977                                  !READS MLE2 DONT CARE MASK EITHER DATA DIAG OR ECC CIE REG
533 ;          1978   ML_REG[14,DONT_CARE]%,
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (3)

;       M 1979      WRT_MASK =
;       M 1980                                          !GENERATE MASK DATA PATTERN USING REGISTER FORCE LO, HI AND IGNORE B
;         1981    .IGNORE or ((not .LO) and (.HI or .TST_PAT))%,
;       M 1982      CLR_MBUS =
;       M 1983                                          !CLEAR MASS BUS RESTORE DRIVE NUMBER
;         1984    CLR = ONE; DRV_NUM = .ML_DUT%,
;         1985    !<BLF/SYNONYM IS_SET = EQL 1 * >
;         1986    !<BLF/SYNONYM IS_NOT_SET = EQL 0 * >
;         1987    !
;         1988    ! DIAGNOSTIC DATA REGISTER MACROS
;         1989    !
;       M 1990      RD_LNG_WRD =
;       M 1991                                  !READ DATA DIAG REGS INTO BIND LOCATIONS
;       M 1992      D1_TEMP = .MLD1;
;       M 1993      D2_TEMP = .MLD2;
;         1994      E2_TEMP = .MLE2%,
;       M 1995      WRT_LNG_WRD =
;       M 1996                                  !LOADS DATA DIAG REG WITH CONTENTS OF BIND LOCATIONS
;       M 1997      MLD1 = .D1_TEMP;
;       M 1998      MLD2 = .D2_TEMP;
;         1999      MLE2 = .E2_TEMP%,
;       M 2000      TIME_OUT_LOOP =
;       M 2001              !WAIT LOOP DIRING MASS BUS TRANSFER
;       M 2002          do
;       M 2003            0
;         2004          until .DRY;%;
;         2005
;         2006    !
;         2007    ! BIND DECLARATIONS
;         2008    !
;         2009
;         2010    bind
;         2011    !
;         2012    ! ERROR DATA MAPPING FORMATS
;         2013    !
;         2014        FMT_1 = uplit (%asciz'%AEXPECTED: %O6%A    READ: %O6%N%N'),
;         2015        FMT_2 = uplit (%asciz'%AGOOD DATA: %O6%A    BAD DATA: %O6%A    XOR: %O6%N%N'),
;         2016        FMT_3 = uplit (%asciz'%ADRIVE SN: %O6%N%N'),
;         2017        FMT_4 = uplit (%asciz'%ABIT IN ERRCR: %O6%N%N'),
;         2018        FMT_5 = uplit (%asciz'%AGOOD NIB DATA: %O2%A    BAD NIB DATA: %O2%A    NIB POS: %O4%N%N'),
;         2019        FMT_6 = uplit (%asciz'%ANIB IN ERROR: %D4%N%N'),
;         2020        FMT_7 = uplit (%asciz'%AFAILED AT: %O6%N%N'),
;         2021        FMT_8 = uplit (%asciz'%AREPLACE ARR MOD: %D2%N%N'),
;         2022        FMT_9 = uplit (%asciz'%AFAILED AT DSA: %O6%N%N'),
;         2023        FMT_10 = uplit (%asciz'%ABIT<15:10>: %B6%A    BIT<9:0>: %B10%N%N'),
;         2024        FMT_11 = uplit (%asciz'%AFAILING REG ADRS: %O6%N%N'),
;         2025        FMT_12 = uplit (%asciz'%AFAILING FUNC: %O6%N%N'),
;         2026        FMT_13 = uplit (%asciz' %AOFF SET CNT FOR NIB : %D2 %A = %D2 %N%N'),
;         2027        FMT_14 = uplit (%asciz'%AWROTE: %D2%A    READ: %D2%N%N'),
;         2028        FMT_15 = uplit (%asciz'%ANIBBLES XFERED BEFORE ERROR: %D3%N'),
;         2029        FMT_16 = uplit (%asciz'%AFAILING REG: %O6%A GOOD DATA: %O6%A BAD DATA: %O6%N%N'),
;         2030        FMT_17 = uplit (%asciz'%N%ADIAGNOSING UNIT %O1%N%N'),
```

```
591 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
592 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (3)
593
594 ;          2031      FMT_18 = uplit (%asciz'%ATIMED OUT DURING MBUS %O2%A FUNC%N%N'),
595 ;          2032      FMT_19 = uplit (%asciz'%ACRC GEN = B: %B6%A A: %B6%A P: %B6%N'),
596 ;          2033      FMT_20 = uplit (%asciz'%ACRC CAL = B: %B6%A A: %B6%A P: %B6%N%N'),
597 ;          2034      FMT_21 = uplit (%asciz'%AFAILED AT PLOG: %D2%A CHANNEL: %D2%N%N'),
598 ;          2035      FMT_22 = uplit (%asciz'%AFAILED AT WRD: %D2%A BIT: %D2%N%N'),
599 ;          2036      FMT_23 = uplit (%asciz'%AREGISTER%S3%AADDRESS%S7%ACONTENTS%N%N'),
600 ;          2037      FMT_24 = uplit (%asciz'%S%T%S5%O6%S4%O16%N'),
601 ;          2038      FMT_25 = uplit (%asciz'%S4%T%A:%S%O6'),
602 ;          2039      FMT_26 = uplit (%asciz'%S4%T%A:%S%D1%D1%D1%D1%N'),
603 ;          2040  !
604 ;          2041  ! ERROR MESSAGE MAPPING FORMATS
605 ;          2042  !
606 ;          2043      ONE_FMT = uplit (%asciz'%T%N'),
607 ;          2044      TWO_FMT = uplit (%asciz'%T%T%N'),
608 ;          2045      THR_FMT = uplit (%asciz'%T%T%T%N'),
609 ;          2046      FOR_FMT = uplit (%asciz'%T%T%T%T%N'),
610 ;          2047      FIV_FMT = uplit (%asciz'%T%T%T%T%T%N'),
611 ;          2048      SIX_FMT = uplit (%asciz'%T%T%T%T%T%T%N'),
612 ;          2049      SEV_FMT = uplit (%asciz'%T%T%T%T%T%T%T%N'),
613 ;          2050      EIG_FMT = uplit (%asciz'%T%T%T%T%T%T%T%T%N'),
614 ;          2051      NIN_FMT = uplit (%asciz'%T%T%T%T%T%T%T%T%T%N'),
615 ;          2052      TEN_FMT = uplit (%asciz'%T%T%T%T%T%T%T%T%T%T%N'),
616 ;          2053      ELV_FMT = uplit (%asciz'%T%T%T%T%T%T%T%T%T%T%N'),
617 ;          2054  !
618 ;          2055  ! DIAGNOSTIC VOCABULARY
619 ;          2056  !
620 ;          2057      !
621 ;          2058      ! WORDS
622 ;          2059      !
623 ;          2060      WRD_1 = uplit (%asciz' GO'),
624 ;          2061      WRD_2 = uplit (%asciz' DRV_RDY'),
625 ;          2062      WRD_3 = uplit (%asciz' ILF'),
626 ;          2063      WRD_4 = uplit (%asciz' OPI'),
627 ;          2064      WRD_5 = uplit (%asciz' BAD'),
628 ;          2065      WRD_6 = uplit (%asciz' GOOD'),
629 ;          2066      WRD_7 = uplit (%asciz' PARITY NOT'),
630 ;          2067      WRD_8 = uplit (%asciz' GENERATED'),
631 ;          2068      WRD_9 = uplit (%asciz' DETECTED'),
632 ;          2069      WRD_10 = uplit (%asciz' ERROR'),
633 ;          2070      WRD_11 = uplit (%asciz' AFTER'),
634 ;          2071      WRD_12 = uplit (%asciz' DURING'),
635 ;          2072      WRD_13 = uplit (%asciz' AT'),
636 ;          2073      WRD_14 = uplit (%asciz' FAILURE'),
637 ;          2074      WRD_15 = uplit (%asciz' ATA'),
638 ;          2075      WRD_16 = uplit (%asciz' ATTN'),
639 ;          2076      WRD_17 = uplit (%asciz' WRITING'),
640 ;          2077      WRD_18 = uplit (%asciz' VV'),
641 ;          2078      WRD_19 = uplit (%asciz' FUNC'),
642 ;          2079      WRD_20 = uplit (%asciz' TRE'),
643 ;          2080      WRD_21 = uplit (%asciz' RMR'),
644 ;          2081      WRD_22 = uplit (%asciz' EXCESSIVE'),
645 ;          2082      WRD_23 = uplit (%asciz' MBUS'),
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (3)

```
;ML4
;
;
;          2083      WRD_24 = uplit (%asciz' DATA'),
;          2084      WRD_25 = uplit (%asciz' CONTINUITY'),
;          2085      WRD_26 = uplit (%asciz' AOE'),
;          2086      WRD_27 = uplit (%asciz' LBT'),
;          2087      WRD_29 = uplit (%asciz' PREMATURLY'),
;          2088      WRD_30 = uplit (%asciz' IAE'),
;          2089      WRD_31 = uplit (%asciz' INCREMENT'),
;          2090      WRD_32 = uplit (%asciz' WITH'),
;          2091      WRD_33 = uplit (%asciz' UV'),
;          2092      WRD_34 = uplit (%asciz' UNS'),
;          2093      WRD_35 = uplit (%asciz' PROM'),
;          2094      WRD_36 = uplit (%asciz' OR'),
;          2095      WRD_37 = uplit (%asciz' SELECT'),
;          2096      WRD_38 = uplit (%asciz' REG'),
;          2097      WRD_39 = uplit (%asciz' UNIQUE'),
;          2098      WRD_40 = uplit (%asciz' 14'),
;          2099      WRD_41 = uplit (%asciz' NIBBLE CNT'),
;          2100      WRD_42 = uplit (%asciz' GTR'),
;          2101      WRD_43 = uplit (%asciz' WHILE'),
;          2102      WRD_44 = uplit (%asciz' TRE'),
;          2103      WRD_45 = uplit (%asciz' INITIAL'),
;          2104      WRD_46 = uplit (%asciz' OFF_SET'),
;          2105      WRD_47 = uplit (%asciz' COUNT'),
;          2106      WRD_48 = uplit (%asciz' DELAY'),
;          2107      WRD_49 = uplit (%asciz' TESTS'),
;          2108      WRD_50 = uplit (%asciz' ADRS'),
;          2109      WRD_51 = uplit (%asciz' COUNTER'),
;          2110      WRD_52 = uplit (%asciz' REG'),
;          2111      WRD_53 = uplit (%asciz' TESTED'),
;          2112      WRD_54 = uplit (%asciz' NIBBLE'),
;          2113      WRD_55 = uplit (%asciz' ALL'),
;          2114      WRD_56 = uplit (%asciz' TEST'),
;          2115      WRD_57 = uplit (%asciz' XFERED'),
;          2116      WRD_58 = uplit (%asciz' NIBBLES'),
;          2117      WRD_59 = uplit (%asciz' SC'),
;          2118      WRD_60 = uplit (%asciz' MULTIPLEXER'),
;          2119      WRD_61 = uplit (%asciz' UNEXPECTED'),
;          2120      WRD_62 = uplit (%asciz' NED'),
;          2121      WRD_63 = uplit (%asciz' ILR'),
;          2122      WRD_64 = uplit (%asciz' CRC'),
;          2123      WRD_65 = uplit (%asciz' SGL'),
;          2124      WRD_67 = uplit (%asciz' ECH'),
;          2125      WRD_68 = uplit (%asciz'UNC'),
;          2126      WRD_69 = uplit (%asciz' BIT'),
;          2127      WRD_70 = uplit (%asciz' CHANNEL'),
;          2128      WRD_71 = uplit (%asciz' LATCH'),
;          2129      WRD_72 = uplit (%asciz' DCK'),
;          2130      WRD_73 = uplit (%asciz' BUS'),
;          2131      WRD_74 = uplit (%asciz' SYNDROME'),
;          2132      WRD_75 = uplit (%asciz' DECODE'),
;          2133      WRD_76 = uplit (%asciz' NOT'),
;          2134      WRD_77 = uplit (%asciz' UNCORRECTABLE'),
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (3)

;          2135      !
;          2136      ! PHRASES
;          2137      !
;          2138      PHR_1 = uplit (%asciz' BIT NOT SET'),
;          2139      PHR_2 = uplit (%asciz' BIT NOT CLR'),
;          2140      PHR_3 = uplit (%asciz' NO RESPONCE AFTER 1.5 US'),
;          2141      PHR_4 = uplit (%asciz' DATA ERRORS'),
;          2142      PHR_5 = uplit (%asciz' BIT SET'),
;          2143      PHR_6 = uplit (%asciz' BIT CLR'),
;          2144      PHR_7 = uplit (%asciz' OF OTHER DRIVES'),
;          2145      PHR_8 = uplit (%asciz' CLASS A'),
;          2146      PHR_9 = uplit (%asciz' CLASS B'),
;          2147      PHR_10 = uplit (%asciz' TO FIND'),
;          2148      PHR_11 = uplit (%asciz' NOT LATCHED'),
;          2149      PHR_12 = uplit (%asciz' SINGLE BIT ERROR'),
;          2150      PHR_13 = uplit (%asciz' MULTIPLE BIT ERROR'),
;          2151      PHR_14 = uplit (%asciz'          REGISTER   DUMP'),
;          2152      PHR_15 = uplit (%asciz'SERIAL #'),
;          2153      !
;          2154      ! FUNCTIONS
;          2155      !
;          2156      FNC_1 = uplit (%asciz' MEM SIZING'),
;          2157      FNC_2 = uplit (%asciz' NOOP'),
;          2158      FNC_3 = uplit (%asciz' DRV'),
;          2159      FNC_4 = uplit (%asciz' WRITE CHECK'),
;          2160      FNC_5 = uplit (%asciz' WRITE'),
;          2161      FNC_6 = uplit (%asciz' READ'),
;          2162      FNC_7 = uplit (%asciz' CLEAR'),
;          2163      FNC_8 = uplit (%asciz' COMP ERROR'),
;          2164      FNC_9 = uplit (%asciz' SYS CLR'),
;          2165      FNC_10 = uplit (%asciz' SEARCH'),
;          2166      FNC_11 = uplit (%asciz' READ-IN-PRESET'),
;          2167      FNC_12 = uplit (%asciz' ILLEGAL'),
;          2168      FNC_13 = uplit (%asciz' ABORT'),
;          2169      FNC_14 = uplit (%asciz' ARR RD WRT'),
;          2170      FNC_15 = uplit (%asciz' GOOD BLK'),
;          2171      FNC_16 = uplit (%asciz' REFRESH'),
;          2172      FNC_17 = uplit (%asciz' ARRAY'),
;          2173      FNC_18 = uplit (%asciz' RAM-BUS'),
;          2174      FNC_19 = uplit (%asciz' OVERFLOW'),
;          2175      FNC_21 = uplit (%asciz' CHK SUM'),
;          2176      FNC_22 = uplit (%asciz' LAST BLK'),
;          2177      FNC_23 = uplit (%asciz' INITIALIZE'),
;          2178      !
;          2179      ! REGISTERS
;          2180      !
;          2181      REG_1 = uplit (%asciz' MLCS1'),
;          2182      REG_2 = uplit (%asciz' MLDS'),
;          2183      REG_3 = uplit (%asciz' MLER'),
;          2184      REG_4 = uplit (%asciz' MLMR'),
;          2185      REG_5 = uplit (%asciz' MLAS'),
;          2186      REG_6 = uplit (%asciz' MLDA'),
```

```
;ML4
;
;          2187      REG_7 = uplit (%asciz' MLDT'),
;          2188      REG_8 = uplit (%asciz' MLPA'),
;          2189      REG_9 = uplit (%asciz' MLSN'),
;          2190      REG_10 = uplit (%asciz' MLE1'),
;          2191      REG_11 = uplit (%asciz' MLE2'),
;          2192      REG_12 = uplit (%asciz' MLD1'),
;          2193      REG_13 = uplit (%asciz' MLD2'),
;          2194      REG_14 = uplit (%asciz' MLEE'),
;          2195      REG_15 = uplit (%asciz' MLEL'),
;          2196      REG_16 = uplit (%asciz' MLPD'),
;          2197      REG_17 = uplit (%asciz' MLCS2'),
;          2198      REG_18 = uplit (%asciz' MLWC'),
;          2199      REG_19 = uplit (%asciz' MLBA'),
;          2200      REG_20 = uplit (%asciz' MLBAI'),
;          2201      REG_21 = uplit (%asciz' MLCS3'),
;          2202      !
;          2203      ! MODULES IN ERROR MESSAGES
;          2204      !
;          2205      ASYNC = uplit (%asciz'ASYNCHRONOUS FAILURE MODULE 7361'),
;          2206      SYNC = uplit (%asciz'SYNCHRONOUS FAILURE MODULE 7362 '),
;          2207      ARR_DAT = uplit (%asciz'ARRAY DATA FAILURE MODULE 7363 '),
;          2208      MEM_ARR = uplit (%asciz'MEMORY ARRAY FAILURE MODULE 7357'),
;          2209      INTER = uplit (%asciz'INTERMEDIATE DIAGNOSTIC MESSAGE'),
;          2210      DATA_LATE = uplit (%asciz'DATA LATE ERROR DURING TRANSFER'),
;          2211      SC_SET = uplit (%asciz'SC BIT SET DURING TRANSFER'),
;          2212      TRBLE_LOOP = uplit (%asciz'TROUBLE SHOOT LOOP ERRORS'),
;          2213      RH_ERROR = uplit (%asciz'RH CONTROLLER ERRORS'),
;          2214      TIME_OUT = uplit (%asciz'DRIVE HUNG AFTER MASS BUS TRANSFER'),
;          2215  !
;          2216  ! DATA DIAGNOSTIC REGISTER SAVE LOCATIONS
;          2217  !
;          2218      D1_TEMP = NIB_SAVE,
;          2219      D2_TEMP = NIB_SAVE [1, 0, 16, 0],
;          2220      E2_TEMP = NIB_SAVE [2, 0, 16, 0];
;          2221
;          2222  !<BLF/PAGE>
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (5)

;          2223  routine LOAD_STACK (STK_PTR, NIB_PTR) : novalue =
;          2224      begin
;          2225
;          2226  !++
;          2227  ! FUNCTIONAL DESCRIPTION:
;          2228  !       LOAD STACK TAKES GOOD NIBBLE DATA
;          2229  !       FOUND IN THE STRUCTURE 'NIB_SAVE'
;          2230  !       AND STORES IT INTO THE STRUCTURE
;          2231  !       'STACK' REWRITING ANY BAD NIBBLE
;          2232  !       'STACK' LOCATIONS WITH GOOD NIBBLE
;          2233  !       DATA
;          2234  !
;          2235  ! FORMAL PARAMETERS:
;          2236  ! STK_PTR
;          2237  !       POINTS TO PRESENT DEPTH OF THE
;          2238  !       'STACK' WHERE PRESENT GOOD NIBBLE
;          2239  !       DATA IS TO BE STORED.
;          2240  !
;          2241  ! NIB_PTR
;          2242  !       POINTS TO CURRENT NIBBLE POSITION BEING
;          2243  !       MANIPULATED.
;          2244  !
;          2245  ! IMPLICIT INPUTS:
;          2246  ! STACK
;          2247  !       VECTOR OF 198 BYTE LOCATIONS WHERE
;          2248  !       GOOD NIBBLE DATA IS STORED
;          2249  !       DURING DIAGNOSTIC MODE READS, AFTER
;          2250  !       BAD NIBBLE LOCATIONS HAVE BEEN
;          2251  !       STRIPPED AWAY.
;          2252  !
;          2253  ! STK_OFF
;          2254  !
;          2255  !    vector of 9 byte LOCATIONS WHICH
;          2256  !       STORES AWAY A BAD NIBBLE OFF_SET
;          2257  !       COUNT FOR EACH NIBBLE POSITION
;          2258  !
;          2259  ! NIB_SAVE
;          2260  !       BLOCK OF 3 WORDS TO STORE THE
;          2261  !       DATA FOUND IN MLD1, MLD2 AND
;          2262  !       MLE2 AFTER A DIAGNOSTIC MODE READ.
;          2263  !
;          2264  ! IMPLICIT OUTPUTS:
;          2265  !       'STACK' LOADED WITH GOOD NIBBLE
;          2266  !       DATA
;          2267  !
;          2268  ! COMPLETETION CODES:   NONE
;          2269  !
;          2270  ! SIDE EFFECTS:         NONE
;          2271  !
;          2272  !--
;          2273
;          2274      case .NIB_PTR from 0 to 9 of              !SELECT NIBBLE DATA TO BE LOADED INTO THE STACK
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (5)

;          2275          set
;          2276
;          2277          [0] :
;          2278              stack [(.STK_PTR - (.STK_OFF [.NIB_PTR]));] = .NIB_SAVE [NIB_0];
;          2279                                                  !LOAD NIBBLE DATA 0 INTO SELECTED STACK LOCATION
;          2280
;          2281          [1] :
;          2282              stack [(.STK_PTR - (.STK_OFF [.NIB_PTR]))] = .NIB_SAVE [NIB_1];
;          2283                                                  !LOAD NIBBLE DATA 1 INTO SELECTED STACK LOCATION
;          2284
;          2285          [2] :
;          2286              stack [(.STK_PTR - (.STK_OFF [.NIB_PTR]))] = .NIB_SAVE [NIB_2];
;          2287                                                  !LOAD NIBBLE DATA 2 INTO SELECTED STACK LOCATION
;          2288
;          2289          [3] :
;          2290              stack [(.STK_PTR - (.STK_OFF [.NIB_PTR]))] = .NIB_SAVE [NIB_3];
;          2291                                                  !LOAD NIBBLE DATA 3 INTO SELECTED STACK LOCATION
;          2292
;          2293          [4] :
;          2294              stack [(.STK_PTR - (.STK_OFF [.NIB_PTR]))] = .NIB_SAVE [NIB_4];
;          2295                                                  !LOAD NIBBLE DATA 4 INTO SELECTED STACK LOCATION
;          2296
;          2297          [5] :
;          2298              stack [(.STK_PTR - (.STK_OFF [.NIB_PTR]))] = .NIB_SAVE [NIB_5];
;          2299                                                  !LOAD NIBBLE DATA 5 INTO SELECTED STACK LOCATION
;          2300
;          2301          [6] :
;          2302              stack [(.STK_PTR - (.STK_OFF [.NIB_PTR]))] = .NIB_SAVE [NIB_6];
;          2303                                                  !LOAD NIBBLE DATA 6 INTO SELECTED STACK LOCATION
;          2304
;          2305          [7] :
;          2306              stack [(.STK_PTR - (.STK_OFF [.NIB_PTR]))] = .NIB_SAVE [NIB_7];
;          2307                                                  !LOAD NIBBLE DATA 7 INTO SELECTED STACK LOCATION
;          2308
;          2309          [8] :
;          2310              stack [(.STK_PTR - (.STK_OFF [.NIB_PTR]))] = .NIB_SAVE [NIB_8];
;          2311                                                  !LOAD NIBBLE DATA 8 INTO SELECTED STACK LOCATION
;          2312
;          2313          [9] :
;          2314              stack [(.STK_PTR - (.STK_OFF [.NIB_PTR]))] = .NIB_SAVE [NIB_9];
;          2315                                                  !LOAD NIBBLE DATA 9 INTO SELECTED STACK LOCATION
;          2316          tes;
;          2317
;          2318      end;
905
906
907
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                        .ASCII  /%AE/
004225  130     120     105             .ASCII  /XPE/
004230  103     124     105             .ASCII  /CTE/
004233  104     072     040             .ASCII  /D: /
004236  045     117     066             .ASCII  /%06/
004241  045     101     040             .ASCII  /%A /
004244  040     040     040             .ASCII  /   /
004247  122     105     101             .ASCII  /REA/
004252  104     072     040             .ASCII  /D: /
004255  045     117     066             .ASCII  /%06/
004260  045     116     045             .ASCII  /%N%/
004263  116     000     000             .ASCII  /N/<00><00>
004266  045     101     107     P.AAB:  .ASCII  /%AG/
004271  117     117     104             .ASCII  /OOD/
004274  040     104     101             .ASCII  / DA/
004277  124     101     072             .ASCII  /TA:/
004302  040     045     117             .ASCII  / %O/
004305  066     045     101             .ASCII  /6%A/
004310  040     040     040             .ASCII  /   /
004313  040     102     101             .ASCII  / BA/
004316  104     040     104             .ASCII  /D D/
004321  101     124     101             .ASCII  /ATA/
004324  072     040     045             .ASCII  /: %/
004327  117     066     045             .ASCII  /06%/
004332  101     040     040             .ASCII  /A  /
004335  040     040     130             .ASCII  /  X/
004340  117     122     072             .ASCII  /OR:/
004343  040     045     117             .ASCII  / %O/
004346  066     045     116             .ASCII  /6%N/
004351  045     116     000             .ASCII  /%N/<00>
004354  045     101     104     P.AAC:  .ASCII  /%AD/
004357  122     111     126             .ASCII  /RIV/
004362  105     040     123             .ASCII  /E S/
004365  116     072     040             .ASCII  /N: /
004370  045     117     066             .ASCII  /%06/
004373  045     116     045             .ASCII  /%N%/
004376  116     000                     .ASCII  /N/<00>
004400  045     101     102     P.AAD:  .ASCII  /%AB/
004403  111     124     040             .ASCII  /IT /
004406  111     116     040             .ASCII  /IN /
004411  105     122     122             .ASCII  /ERR/
004414  117     122     072             .ASCII  /OR:/
004417  040     045     117             .ASCII  / %O/
004422  066     045     116             .ASCII  /6%N/
004425  045     116     000             .ASCII  /%N/<00>
004430  045     101     107     P.AAE:  .ASCII  /%AG/
004433  117     117     104             .ASCII  /OOD/
004436  040     116     111             .ASCII  / NI/
004441  102     040     104             .ASCII  /B D/
004444  101     124     101             .ASCII  /ATA/
004447  072     040     045             .ASCII  /: %/
004452  117     062     045             .ASCII  /02%/
```

```
                                        ;ML4
                                        ;
 004455  101  040  040          .ASCII  /A  /
 004460  040  040  102          .ASCII  /  B/
 004463  101  104  040          .ASCII  /AD /
 004466  116  111  102          .ASCII  /NIB/
 004471  040  104  101          .ASCII  / DA/
 004474  124  101  072          .ASCII  /TA:/
 004477  040  045  117          .ASCII  / %O/
 004502  062  045  101          .ASCII  /2%A/
 004505  040  040  040          .ASCII  /   /
 004510  040  116  111          .ASCII  / NI/
 004513  102  040  120          .ASCII  /B P/
 004516  117  123  072          .ASCII  /OS:/
 004521  040  045  117          .ASCII  / %O/
 004524  064  045  116          .ASCII  /4%N/
 004527  045  116  000          .ASCII  /%N/<00>
 004532  045  101  116  P.AAF:  .ASCII  /%AN/
 004535  111  102  040          .ASCII  /IB /
 004540  111  116  040          .ASCII  /IN /
 004543  105  122  122          .ASCII  /ERR/
 004546  117  122  072          .ASCII  /OR:/
 004551  040  045  104          .ASCII  / %D/
 004554  064  045  116          .ASCII  /4%N/
 004557  045  116  000          .ASCII  /%N/<00>
 004562  045  101  106  P.AAG:  .ASCII  /%AF/
 004565  101  111  114          .ASCII  /AIL/
 004570  105  104  040          .ASCII  /ED /
 004573  101  124  072          .ASCII  /AT:/
 004576  040  045  117          .ASCII  / %O/
 004601  066  045  116          .ASCII  /6%N/
 004604  045  116  000          .ASCII  /%N/<00>
 004607  000                    .ASCII  <00>
 004610  045  101  122  P.AAH:  .ASCII  /%AR/
 004613  105  120  114          .ASCII  /EPL/
 004616  101  103  105          .ASCII  /ACE/
 004621  040  101  122          .ASCII  / AR/
 004624  122  040  115          .ASCII  /R M/
 004627  117  104  072          .ASCII  /OD:/
 004632  040  045  104          .ASCII  / %D/
 004635  062  045  116          .ASCII  /2%N/
 004640  045  116  000          .ASCII  /%N/<00>
 004643  000                    .ASCII  <00>
 004644  045  101  106  P.AAI:  .ASCII  /%AF/
 004647  101  111  114          .ASCII  /AIL/
 004652  105  104  040          .ASCII  /ED /
 004655  101  124  040          .ASCII  /AT /
 004660  104  123  101          .ASCII  /DSA/
 004663  072  040  045          .ASCII  /: %/
 004666  117  066  045          .ASCII  /O6%/
 004671  116  045  116          .ASCII  /N%N/
 004674  000  000                .ASCII  <00><00>
 004676  045  101  102  P.AAJ:  .ASCII  /%AB/
 004701  111  124  074          .ASCII  /IT</
```

```
1021                                    ;ML4
1022                                    ;
1023
1024 004704    061     065     072             .ASCII  /15:/
1025 004707    061     060     076             .ASCII  /10>/
1026 004712    072     040     045             .ASCII  /: %/
1027 004715    102     066     045             .ASCII  /B6%/
1028 004720    101     040     040             .ASCII  /A  /
1029 004723    040     040     102             .ASCII  /  B/
1030 004726    111     124     074             .ASCII  /IT</
1031 004731    071     072     060             .ASCII  /9:0/
1032 004734    076     072     040             .ASCII  />: /
1033 004737    045     102     061             .ASCII  /%B1/
1034 004742    060     045     116             .ASCII  /0%N/
1035 004745    045     116     000             .ASCII  /%N/<00>
1036 004750    045     101     106     P.AAK:  .ASCII  /%AF/
1037 004753    101     111     114             .ASCII  /AIL/
1038 004756    111     116     107             .ASCII  /ING/
1039 004761    040     122     105             .ASCII  / RE/
1040 004764    107     040     101             .ASCII  /G A/
1041 004767    104     122     123             .ASCII  /DRS/
1042 004772    072     040     045             .ASCII  /: %/
1043 004775    117     066     045             .ASCII  /O6%/
1044 005000    116     045     116             .ASCII  /N%N/
1045 005003    000                             .ASCII  <00>
1046 005004    045     101     106     P.AAL:  .ASCII  /%AF/
1047 005007    101     111     114             .ASCII  /AIL/
1048 005012    111     116     107             .ASCII  /ING/
1049 005015    040     106     125             .ASCII  / FU/
1050 005020    116     103     072             .ASCII  /NC:/
1051 005023    040     045     117             .ASCII  / %O/
1052 005026    066     045     116             .ASCII  /6%N/
1053 005031    045     116     000             .ASCII  /%N/<00>
1054 005034    040     045     101     P.AAM:  .ASCII  / %A/
1055 005037    117     106     106             .ASCII  /OFF/
1056 005042    137     123     105             .ASCII  / SE/
1057 005045    124     040     103             .ASCII  /T C/
1058 005050    116     124     040             .ASCII  /NT /
1059 005053    106     117     122             .ASCII  /FOR/
1060 005056    040     116     111             .ASCII  / NI/
1061 005061    102     040     072             .ASCII  /B :/
1062 005064    040     045     104             .ASCII  / %D/
1063 005067    062     040     045             .ASCII  /2 %/
1064 005072    101     040     075             .ASCII  /A =/
1065 005075    040     045     104             .ASCII  / %D/
1066 005100    062     040     045             .ASCII  /2 %/
1067 005103    116     045     116             .ASCII  /N%N/
1068 005106    000     000                     .ASCII  <00><00>
1069 005110    045     101     127     P.AAN:  .ASCII  /%AW/
1070 005113    122     117     124             .ASCII  /ROT/
1071 005116    105     072     040             .ASCII  /E: /
1072 005121    045     104     062             .ASCII  /%D2/
1073 005124    045     101     040             .ASCII  /%A /
1074 005127    040     040     040             .ASCII  /   /
1075 005132    122     105     101             .ASCII  /REA/
```

```
1077                                    ;ML4
1078                                    ;
1079
1080 005135  104  072  040             .ASCII  /D: /
1081 005140  045  104  062             .ASCII  /%D2/
1082 005143  045  116  045             .ASCII  /%N%/
1083 005146  116  000                  .ASCII  /N/<00>
1084 005150  045  101  116  P.AAO:     .ASCII  /%AN/
1085 005153  111  102  102             .ASCII  /IBB/
1086 005156  114  105  123             .ASCII  /LES/
1087 005161  040  130  106             .ASCII  / XF/
1088 005164  105  122  105             .ASCII  /ERE/
1089 005167  104  040  102             .ASCII  /D B/
1090 005172  105  106  117             .ASCII  /EFO/
1091 005175  122  105  040             .ASCII  /RE /
1092 005200  105  122  122             .ASCII  /ERR/
1093 005203  117  122  072             .ASCII  /OR:/
1094 005206  040  045  104             .ASCII  / %D/
1095 005211  063  045  116             .ASCII  /3%N/
1096 005214  000  000                  .ASCII  <00><00>
1097 005216  045  101  106  P.AAP:     .ASCII  /%AF/
1098 005221  101  111  114             .ASCII  /AIL/
1099 005224  111  116  107             .ASCII  /ING/
1100 005227  040  122  105             .ASCII  / RE/
1101 005232  107  072  040             .ASCII  /G: /
1102 005235  045  117  066             .ASCII  /%06/
1103 005240  045  101  040             .ASCII  /%A /
1104 005243  107  117  117             .ASCII  /GOO/
1105 005246  104  040  104             .ASCII  /D D/
1106 005251  101  124  101             .ASCII  /ATA/
1107 005254  072  040  045             .ASCII  /: %/
1108 005257  117  066  045             .ASCII  /06%/
1109 005262  101  040  102             .ASCII  /A B/
1110 005265  101  104  040             .ASCII  /AD /
1111 005270  104  101  124             .ASCII  /DAT/
1112 005273  101  072  040             .ASCII  /A: /
1113 005276  045  117  066             .ASCII  /%06/
1114 005301  045  116  045             .ASCII  /%N%/
1115 005304  116  000                  .ASCII  /N/<00>
1116 005306  045  116  045  P.AAQ:     .ASCII  /%N%/
1117 005311  101  104  111             .ASCII  /ADI/
1118 005314  101  107  116             .ASCII  /AGN/
1119 005317  117  123  111             .ASCII  /OSI/
1120 005322  116  107  040             .ASCII  /NG /
1121 005325  125  116  111             .ASCII  /UNI/
1122 005330  124  040  045             .ASCII  /T %/
1123 005333  117  061  045             .ASCII  /01%/
1124 005336  116  045  116             .ASCII  /N%N/
1125 005341  000                       .ASCII  <00>
1126 005342  045  101  124  P.AAR:     .ASCII  /%AT/
1127 005345  111  115  105             .ASCII  /IME/
1128 005350  104  040  117             .ASCII  /D O/
1129 005353  125  124  040             .ASCII  /UT /
1130 005356  104  125  122             .ASCII  /DUR/
1131 005361  111  116  107             .ASCII  /ING/
```

```
1133                                    ;ML4
1134                                    :
1135
1136 005364  040  115  102             .ASCII  / MB/
1137 005367  125  123  040             .ASCII  /US /
1138 005372  045  117  062             .ASCII  /%02/
1139 005375  045  101  040             .ASCII  /%A /
1140 005400  106  125  116             .ASCII  /FUN/
1141 005403  103  045  116             .ASCII  /C%N/
1142 005406  045  116  000             .ASCII  /%N/<00>
1143 005411  000                       .ASCII  <00>
1144 005412  045  101  103  P.AAS:     .ASCII  /%AC/
1145 005415  122  103  040             .ASCII  /RC /
1146 005420  107  105  116             .ASCII  /GEN/
1147 005423  040  075  040             .ASCII  / = /
1148 005426  102  072  040             .ASCII  /B: /
1149 005431  045  102  066             .ASCII  /%B6/
1150 005434  045  101  040             .ASCII  /%A /
1151 005437  101  072  040             .ASCII  /A: /
1152 005442  045  102  066             .ASCII  /%B6/
1153 005445  045  101  040             .ASCII  /%A /
1154 005450  120  072  040             .ASCII  /P: /
1155 005453  045  102  066             .ASCII  /%B6/
1156 005456  045  116  000             .ASCII  /%N/<00>
1157 005461  000                       .ASCII  <00>
1158 005462  045  101  103  P.AAT:     .ASCII  /%AC/
1159 005465  122  103  040             .ASCII  /RC /
1160 005470  103  101  114             .ASCII  /CAL/
1161 005473  040  075  040             .ASCII  / = /
1162 005476  102  072  040             .ASCII  /B: /
1163 005501  045  102  066             .ASCII  /%B6/
1164 005504  045  101  040             .ASCII  /%A /
1165 005507  101  072  040             .ASCII  /A: /
1166 005512  045  102  066             .ASCII  /%B6/
1167 005515  045  101  040             .ASCII  /%A /
1168 005520  120  072  040             .ASCII  /P: /
1169 005523  045  102  066             .ASCII  /%B6/
1170 005526  045  116  045             .ASCII  /%N%/
1171 005531  116  000  000             .ASCII  /N/<00><00>
1172 005534  045  101  106  P.AAU:     .ASCII  /%AF/
1173 005537  101  111  114             .ASCII  /AIL/
1174 005542  105  104  040             .ASCII  /ED /
1175 005545  101  124  040             .ASCII  /AT /
1176 005550  120  114  117             .ASCII  /PLO/
1177 005553  107  072  040             .ASCII  /G: /
1178 005556  045  104  062             .ASCII  /%D2/
1179 005561  045  101  040             .ASCII  /%A /
1180 005564  103  110  101             .ASCII  /CHA/
1181 005567  116  116  105             .ASCII  /NNE/
1182 005572  114  072  040             .ASCII  /L: /
1183 005575  045  104  062             .ASCII  /%D2/
1184 005600  045  116  045             .ASCII  /%N%/
1185 005603  116  000  000             .ASCII  /N/<00><00>
1186 005606  045  101  106  P.AAV:     .ASCII  /%AF/
1187 005611  101  111  114             .ASCII  /AIL/
```

```
1189                                    ;ML4
1190                                    ;
1191
1192 005614    105     104     040             .ASCII  /ED /
1193 005617    101     124     040             .ASCII  /AT /
1194 005622    127     122     104             .ASCII  /WRD/
1195 005625    072     040     045             .ASCII  /: %/
1196 005630    104     062     045             .ASCII  /D2%/
1197 005633    101     040     102             .ASCII  /A B/
1198 005636    111     124     072             .ASCII  /IT:/
1199 005641    040     045     104             .ASCII  / %D/
1200 005644    062     045     116             .ASCII  /2%N/
1201 005647    045     116     000             .ASCII  /%N/<00>
1202 005652    045     101     122     P.AAW:  .ASCII  /%AR/
1203 005655    105     107     111             .ASCII  /EGI/
1204 005660    123     124     105             .ASCII  /STE/
1205 005663    122     045     123             .ASCII  /R%S/
1206 005666    063     045     101             .ASCII  /3%A/
1207 005671    101     104     104             .ASCII  /ADD/
1208 005674    122     105     123             .ASCII  /RES/
1209 005677    123     045     123             .ASCII  /S%S/
1210 005702    067     045     101             .ASCII  /7%A/
1211 005705    103     117     116             .ASCII  /CON/
1212 005710    124     105     116             .ASCII  /TEN/
1213 005713    124     123     045             .ASCII  /TS%/
1214 005716    116     045     116             .ASCII  /N%N/
1215 005721    000                             .ASCII  <00>
1216 005722    045     123     045     P.AAX:  .ASCII  /%S%/
1217 005725    124     045     123             .ASCII  /T%S/
1218 005730    065     045     117             .ASCII  /5%O/
1219 005733    066     045     123             .ASCII  /6%S/
1220 005736    064     045     117             .ASCII  /4%O/
1221 005741    061     066     045             .ASCII  /16%/
1222 005744    116     000                     .ASCII  /N/<00>
1223 005746    045     123     064     P.AAY:  .ASCII  /%S4/
1224 005751    045     124     045             .ASCII  /%T%/
1225 005754    101     072     045             .ASCII  /A:%/
1226 005757    123     045     117             .ASCII  /S%O/
1227 005762    066     000                     .ASCII  /6/<00>
1228 005764    045     123     064     P.AAZ:  .ASCII  /%S4/
1229 005767    045     124     045             .ASCII  /%T%/
1230 005772    101     072     045             .ASCII  /A:%/
1231 005775    123     045     104             .ASCII  /S%D/
1232 006000    061     045     104             .ASCII  /1%D/
1233 006003    061     045     104             .ASCII  /1%D/
1234 006006    061     045     104             .ASCII  /1%D/
1235 006011    061     045     116             .ASCII  /1%N/
1236 006014    000     000                     .ASCII  <00><00>
1237 006016    045     124     045     P.ABA:  .ASCII  /%T%/
1238 006021    116     000     000             .ASCII  /N/<00><00>
1239 006024    045     124     045     P.ABB:  .ASCII  /%T%/
1240 006027    124     045     116             .ASCII  /T%N/
1241 006032    000     000                     .ASCII  <00><00>
1242 006034    045     124     045     P.ABC:  .ASCII  /%T%/
1243 006037    124     045     124             .ASCII  /T%T/
```

```
1245                                    ;ML4
1246                                    ;
1247
1248 006042  045  116  000             .ASCII  /%N/<00>
1249 006045  000                        .ASCII  <00>
1250 006046  045  124  045  P.ABD:     .ASCII  /%T%/
1251 006051  124  045  124              .ASCII  /T%T/
1252 006054  045  124  045              .ASCII  /%T%/
1253 006057  116  000  000              .ASCII  /N/<00><00>
1254 006062  045  124  045  P.ABE:     .ASCII  /%T%/
1255 006065  124  045  124              .ASCII  /T%T/
1256 006070  045  124  045              .ASCII  /%T%/
1257 006073  124  045  116              .ASCII  /T%N/
1258 006076  000  000                   .ASCII  <00><00>
1259 006100  045  124  045  P.ABF:     .ASCII  /%T%/
1260 006103  124  045  124              .ASCII  /T%T/
1261 006106  045  124  045              .ASCII  /%T%/
1262 006111  124  045  124              .ASCII  /T%T/
1263 006114  045  116  000              .ASCII  /%N/<00>
1264 006117  000                        .ASCII  <00>
1265 006120  045  124  045  P.ABG:     .ASCII  /%T%/
1266 006123  124  045  124              .ASCII  /T%T/
1267 006126  045  124  045              .ASCII  /%T%/
1268 006131  124  045  124              .ASCII  /T%T/
1269 006134  045  124  045              .ASCII  /%T%/
1270 006137  116  000  000              .ASCII  /N/<00><00>
1271 006142  045  124  045  P.ABH:     .ASCII  /%T%/
1272 006145  124  045  124              .ASCII  /T%T/
1273 006150  045  124  045              .ASCII  /%T%/
1274 006153  124  045  124              .ASCII  /T%T/
1275 006156  045  124  045              .ASCII  /%T%/
1276 006161  124  045  116              .ASCII  /T%N/
1277 006164  000  000                   .ASCII  <00><00>
1278 006166  045  124  045  P.ABI:     .ASCII  /%T%/
1279 006171  124  045  124              .ASCII  /T%T/
1280 006174  045  124  045              .ASCII  /%T%/
1281 006177  124  045  124              .ASCII  /T%T/
1282 006202  045  124  045              .ASCII  /%T%/
1283 006205  124  045  124              .ASCII  /T%T/
1284 006210  045  116  000              .ASCII  /%N/<00>
1285 006213  000                        .ASCII  <00>
1286 006214  045  124  045  P.ABJ:     .ASCII  /%T%/
1287 006217  124  045  124              .ASCII  /T%T/
1288 006222  045  124  045              .ASCII  /%T%/
1289 006225  124  045  124              .ASCII  /T%T/
1290 006230  045  124  045              .ASCII  /%T%/
1291 006233  124  045  124              .ASCII  /T%T/
1292 006236  045  124  045              .ASCII  /%T%/
1293 006241  116  000  000              .ASCII  /N/<00><00>
1294 006244  045  124  045  P.ABK:     .ASCII  /%T%/
1295 006247  124  045  124              .ASCII  /T%T/
1296 006252  045  124  045              .ASCII  /%T%/
1297 006255  124  045  124              .ASCII  /T%T/
1298 006260  045  124  045              .ASCII  /%T%/
1299 006263  124  045  124              .ASCII  /T%T/
```

14-May-1981 13:48:54 TOPS
14-May-1981 13:05:08 PA:<

```
1301                                    ;ML4
1302                                    ;
1303
1304 006266  045  124  045             .ASCII  /%T%/
1305 006271  124  045  116             .ASCII  /T%N/
1306 006274  000  000                  .ASCII  <00><00>
1307 006276  040  107  117  P.ABL:     .ASCII  / GO/
1308 006301  000                       .ASCII  <00>
1309 006302  040  104  122  P.ABM:     .ASCII  / DR/
1310 006305  126  137  122             .ASCII  /V_R/
1311 006310  104  131  000             .ASCII  /DY/<00>
1312 006313  000                       .ASCII  <00>
1313 006314  040  111  114  P.ABN:     .ASCII  / IL/
1314 006317  106  000  000             .ASCII  /F/<00><00>
1315 006322  040  117  120  P.ABO:     .ASCII  / OP/
1316 006325  111  000  000             .ASCII  /I/<00><00>
1317 006330  040  102  101  P.ABP:     .ASCII  / BA/
1318 006333  104  000  000             .ASCII  /D/<00><00>
1319 006336  040  107  117  P.ABQ:     .ASCII  / GO/
1320 006341  117  104  000             .ASCII  /OD/<00>
1321 006344  040  120  101  P.ABR:     .ASCII  / PA/
1322 006347  122  111  124             .ASCII  /RIT/
1323 006352  131  040  116             .ASCII  /Y N/
1324 006355  117  124  000             .ASCII  /OT/<00>
1325 006360  040  107  105  P.ABS:     .ASCII  / GE/
1326 006363  116  105  122             .ASCII  /NER/
1327 006366  101  124  105             .ASCII  /ATE/
1328 006371  104  000  000             .ASCII  /D/<00><00>
1329 006374  040  104  105  P.ABT:     .ASCII  / DE/
1330 006377  124  105  103             .ASCII  /TEC/
1331 006402  124  105  104             .ASCII  /TED/
1332 006405  000                       .ASCII  <00>
1333 006406  040  105  122  P.ABU:     .ASCII  / ER/
1334 006411  122  117  122             .ASCII  /ROR/
1335 006414  000  000                  .ASCII  <00><00>
1336 006416  040  101  106  P.ABV:     .ASCII  / AF/
1337 006421  124  105  122             .ASCII  /TER/
1338 006424  000  000                  .ASCII  <00><00>
1339 006426  040  104  125  P.ABW:     .ASCII  / DU/
1340 006431  122  111  116             .ASCII  /RIN/
1341 006434  107  000                  .ASCII  /G/<00>
1342 006436  040  101  124  P.ABX:     .ASCII  / AT/
1343 006441  000                       .ASCII  <00>
1344 006442  040  106  101  P.ABY:     .ASCII  / FA/
1345 006445  111  114  125             .ASCII  /ILU/
1346 006450  122  105  000             .ASCII  /RE/<00>
1347 006453  000                       .ASCII  <00>
1348 006454  040  101  124  P.ABZ:     .ASCII  / AT/
1349 006457  101  000  000             .ASCII  /A/<00><00>
1350 006462  040  101  124  P.ACA:     .ASCII  / AT/
1351 006465  124  116  000             .ASCII  /TN/<00>
1352 006470  040  127  122  P.ACB:     .ASCII  / WR/
1353 006473  111  124  111             .ASCII  /ITI/
1354 006476  116  107  000             .ASCII  /NG/<00>
1355 006501  000                       .ASCII  <00>
```

```
1357                                    ;ML4
1358                                    ;
1359
1360 006502  040  126  126  P.ACC:  .ASCII  / VV/
1361 006505  000                     .ASCII  <00>
1362 006506  040  106  125  P.ACD:  .ASCII  / FU/
1363 006511  116  103  000          .ASCII  /NC/<00>
1364 006514  040  124  122  P.ACE:  .ASCII  / TR/
1365 006517  105  000  000          .ASCII  /E/<00><00>
1366 006522  040  122  115  P.ACF:  .ASCII  / RM/
1367 006525  122  000  000          .ASCII  /R/<00><00>
1368 006530  040  105  130  P.ACG:  .ASCII  / EX/
1369 006533  103  105  123          .ASCII  /CES/
1370 006536  123  111  126          .ASCII  /SIV/
1371 006541  105  000  000          .ASCII  /E/<00><00>
1372 006544  040  115  102  P.ACH:  .ASCII  / MB/
1373 006547  125  123  000          .ASCII  /US/<00>
1374 006552  040  104  101  P.ACI:  .ASCII  / DA/
1375 006555  124  101  000          .ASCII  /TA/<00>
1376 006560  040  103  117  P.ACJ:  .ASCII  / CO/
1377 006563  116  124  111          .ASCII  /NTI/
1378 006566  116  125  111          .ASCII  /NUI/
1379 006571  124  131  000          .ASCII  /TY/<00>
1380 006574  040  101  117  P.ACK:  .ASCII  / AO/
1381 006577  105  000  000          .ASCII  /E/<00><00>
1382 006602  040  114  102  P.ACL:  .ASCII  / LB/
1383 006605  124  000  000          .ASCII  /T/<00><00>
1384 006610  040  120  122  P.ACM:  .ASCII  / PR/
1385 006613  105  115  101          .ASCII  /EMA/
1386 006616  124  125  122          .ASCII  /TUR/
1387 006621  114  131  000          .ASCII  /LY/<00>
1388 006624  040  111  101  P.ACN:  .ASCII  / IA/
1389 006627  105  000  000          .ASCII  /E/<00><00>
1390 006632  040  111  116  P.ACO:  .ASCII  / IN/
1391 006635  103  122  105          .ASCII  /CRE/
1392 006640  115  105  116          .ASCII  /MEN/
1393 006643  124  000  000          .ASCII  /T/<00><00>
1394 006646  040  127  111  P.ACP:  .ASCII  / WI/
1395 006651  124  110  000          .ASCII  /TH/<00>
1396 006654  040  125  126  P.ACQ:  .ASCII  / UV/
1397 006657  000                     .ASCII  <00>
1398 006660  040  125  116  P.ACR:  .ASCII  / UN/
1399 006663  123  000  000          .ASCII  /S/<00><00>
1400 006666  040  120  122  P.ACS:  .ASCII  / PR/
1401 006671  117  115  000          .ASCII  /OM/<00>
1402 006674  040  117  122  P.ACT:  .ASCII  / OR/
1403 006677  000                     .ASCII  <00>
1404 006700  040  123  105  P.ACU:  .ASCII  / SE/
1405 006703  114  105  103          .ASCII  /LEC/
1406 006706  124  000               .ASCII  /T/<00>
1407 006710  040  122  105  P.ACV:  .ASCII  / RE/
1408 006713  107  000  000          .ASCII  /G/<00><00>
1409 006716  040  125  116  P.ACW:  .ASCII  / UN/
1410 006721  111  121  125          .ASCII  /IQU/
1411 006724  105  000               .ASCII  /E/<00>
```

```
1413                                    ;ML4
1414                                    ;
1415
1416 006726  040  061  064  P.ACX:  .ASCII  / 14/
1417 006731  000                    .ASCII  <00>
1418 006732  040  116  111  P.ACY:  .ASCII  / NI/
1419 006735  102  102  114          .ASCII  /BBL/
1420 006740  105  040  103          .ASCII  /E C/
1421 006743  116  124  000          .ASCII  /NT/<00>
1422 006746  040  107  124  P.ACZ:  .ASCII  / GT/
1423 006751  122  000  000          .ASCII  /R/<00><00>
1424 006754  040  127  110  P.ADA:  .ASCII  / WH/
1425 006757  111  114  105          .ASCII  /ILE/
1426 006762  000  000               .ASCII  <00><00>
1427 006764  040  124  122  P.ADB:  .ASCII  / TR/
1428 006767  105  000  000          .ASCII  /E/<00><00>
1429 006772  040  111  116  P.ADC:  .ASCII  / IN/
1430 006775  111  124  111          .ASCII  /ITI/
1431 007000  101  114  000          .ASCII  /AL/<00>
1432 007003  000                    .ASCII  <00>
1433 007004  040  117  106  P.ADD:  .ASCII  / OF/
1434 007007  106  137  123          .ASCII  /F_S/
1435 007012  105  124  000          .ASCII  /ET/<00>
1436 007015  000                    .ASCII  <00>
1437 007016  040  103  117  P.ADE:  .ASCII  / CO/
1438 007021  125  116  124          .ASCII  /UNT/
1439 007024  000  000               .ASCII  <00><00>
1440 007026  040  104  105  P.ADF:  .ASCII  / DE/
1441 007031  114  101  131          .ASCII  /LAY/
1442 007034  000  000               .ASCII  <00><00>
1443 007036  040  124  105  P.ADG:  .ASCII  / TE/
1444 007041  123  124  123          .ASCII  /STS/
1445 007044  000  000               .ASCII  <00><00>
1446 007046  040  101  104  P.ADH:  .ASCII  / AD/
1447 007051  122  123  000          .ASCII  /RS/<00>
1448 007054  040  103  117  P.ADI:  .ASCII  / CO/
1449 007057  125  116  124          .ASCII  /UNT/
1450 007062  105  122  000          .ASCII  /ER/<00>
1451 007065  000                    .ASCII  <00>
1452 007066  040  122  105  P.ADJ:  .ASCII  / RE/
1453 007071  107  000  000          .ASCII  /G/<00><00>
1454 007074  040  124  105  P.ADK:  .ASCII  / TE/
1455 007077  123  124  105          .ASCII  /STE/
1456 007102  104  000               .ASCII  /D/<00>
1457 007104  040  116  111  P.ADL:  .ASCII  / NI/
1458 007107  102  102  114          .ASCII  /BBL/
1459 007112  105  000               .ASCII  /E/<00>
1460 007114  040  101  114  P.ADM:   ASCII  / AL/
1461 007117  114  000  000          .ASCII  /L/<00><00>
1462 007122  040  124  105  P.ADN:  .ASCII  / TE/
1463 007125  123  124  000          .ASCII  /ST/<00>
1464 007130  040  130  106  P.ADO:  .ASCII  / XF/
1465 007133  105  122  105          .ASCII  /ERE/
1466 007136  104  000               .ASCII  /D/<00>
1467 007140  040  116  111  P.ADP:  .ASCII  / NI/
```

```
1469                                    ;ML4
1470                                    ;
1471
1472 007143    102     102     114             .ASCII  /BBL/
1473 007146    105     123     000             .ASCII  /ES/<00>
1474 007151    000                             .ASCII  <00>
1475 007152    040     123     103     P.ADQ:  .ASCII  / SC/
1476 007155    000                             .ASCII  <00>
1477 007156    040     115     125     P.ADR:  .ASCII  / MU/
1478 007161    114     124     111             .ASCII  /LTI/
1479 007164    120     114     105             .ASCII  /PLE/
1480 007167    130     105     122             .ASCII  /XER/
1481 007172    000     000                     .ASCII  <00><00>
1482 007174    040     125     116     P.ADS:  .ASCII  / UN/
1483 007177    105     130     120             .ASCII  /EXP/
1484 007202    105     103     124             .ASCII  /ECT/
1485 007205    105     104     000             .ASCII  /ED/<00>
1486 007210    040     116     105     P.ADT:  .ASCII  / NE/
1487 007213    104     000     000             .ASCII  /D/<00><00>
1488 007216    040     111     114     P.ADU:  .ASCII  / IL/
1489 007221    122     000     000             .ASCII  /R/<00><00>
1490 007224    040     103     122     P.ADV:  .ASCII  / CR/
1491 007227    103     000     000             .ASCII  /C/<00><00>
1492 007232    040     123     107     P.ADW:  .ASCII  / SG/
1493 007235    114     000     000             .ASCII  /L/<00><00>
1494 007240    040     105     103     P.ADX:  .ASCII  / EC/
1495 007243    110     000     000             .ASCII  /H/<00><00>
1496 007246    125     116     103     P.ADY:  .ASCII  /UNC/
1497 007251    000                             .ASCII  <00>
1498 007252    040     102     111     P.ADZ:  .ASCII  / BI/
1499 007255    124     000     000             .ASCII  /T/<00><00>
1500 007260    040     103     110     P.AEA:  .ASCII  / CH/
1501 007263    101     116     116             .ASCII  /ANN/
1502 007266    105     114     000             .ASCII  /EL/<00>
1503 007271    000                             .ASCII  <00>
1504 007272    040     114     101     P.AEB:  .ASCII  / LA/
1505 007275    124     103     110             .ASCII  /TCH/
1506 007300    000     000                     .ASCII  <00><00>
1507 007302    040     104     103     P.AEC:  .ASCII  / DC/
1508 007305    113     000     000             .ASCII  /K/<00><00>
1509 007310    040     102     125     P.AED:  .ASCII  / BU/
1510 007313    123     000     000             .ASCII  /S/<00><00>
1511 007316    040     123     131     P.AEE:  .ASCII  / SY/
1512 007321    116     104     122             .ASCII  /NDR/
1513 007324    117     116     105             .ASCII  /ONE/
1514 007327    000                             .ASCII  <00>
1515 007330    040     104     105     P.AEF:  .ASCII  / DE/
1516 007333    103     117     104             .ASCII  /COD/
1517 007336    105     000                     .ASCII  /E/<00>
1518 007340    040     116     117     P.AEG:  .ASCII  / NO/
1519 007343    124     000     000             .ASCII  /T/<00><00>
1520 007346    040     125     116     P.AEH:  .ASCII  / UN/
1521 007351    103     117     122             .ASCII  /COR/
1522 007354    122     105     103             .ASCII  /REC/
1523 007357    124     101     102             .ASCII  /TAB/
```

14-May-1981 13:48:54 TOPS
14-May-1981 13:05:08 PA:<

```
1525                                    ;ML4
1526                                    ;
1527
1528 007362  114  105  000             .ASCII  /LE/<00>
1529 007365  000                        .ASCII  <00>
1530 007366  040  102  111  P.AEI:     .ASCII  / BI/
1531 007371  124  040  116             .ASCII  /T N/
1532 007374  117  124  040             .ASCII  /OT /
1533 007377  123  105  124             .ASCII  /SET/
1534 007402  000  000                  .ASCII  <00><00>
1535 007404  040  102  111  P.AEJ:     .ASCII  / BI/
1536 007407  124  040  116             .ASCII  /T N/
1537 007412  117  124  040             .ASCII  /OT /
1538 007415  103  114  122             .ASCII  /CLR/
1539 007420  000  000                  .ASCII  <00><00>
1540 007422  040  116  117  P.AEK:     .ASCII  / NO/
1541 007425  040  122  105             .ASCII  / RE/
1542 007430  123  120  117             .ASCII  /SPO/
1543 007433  116  103  105             .ASCII  /NCE/
1544 007436  040  101  106             .ASCII  / AF/
1545 007441  124  105  122             .ASCII  /TER/
1546 007444  040  061  056             .ASCII  / 1./
1547 007447  065  040  125             .ASCII  /5 U/
1548 007452  123  000                  .ASCII  /S/<00>
1549 007454  040  104  101  P.AEL:     .ASCII  / DA/
1550 007457  124  101  040             .ASCII  /TA /
1551 007462  105  122  122             .ASCII  /ERR/
1552 007465  117  122  123             .ASCII  /ORS/
1553 007470  000  000                  .ASCII  <00><00>
1554 007472  040  102  111  P.AEM:     .ASCII  / BI/
1555 007475  124  040  123             .ASCII  /T S/
1556 007500  105  124  000             .ASCII  /ET/<00>
1557 007503  000                       .ASCII  <00>
1558 007504  040  102  111  P.AEN:     .ASCII  / BI/
1559 007507  124  040  103             .ASCII  /T C/
1560 007512  114  122  000             .ASCII  /LR/<00>
1561 007515  000                       .ASCII  <00>
1562 007516  040  117  106  P.AEO:     .ASCII  / OF/
1563 007521  040  117  124             .ASCII  / OT/
1564 007524  110  105  122             .ASCII  /HER/
1565 007527  040  104  122             .ASCII  / DR/
1566 007532  111  126  105             .ASCII  /IVE/
1567 007535  123  000  000             .ASCII  /S/<00><00>
1568 007540  040  103  114  P.AEP:     .ASCII  / CL/
1569 007543  101  123  123             .ASCII  /ASS/
1570 007546  040  101  000             .ASCII  / A/<00>
1571 007551  000                       .ASCII  <00>
1572 007552  040  103  114  P.AEQ:     .ASCII  / CL/
1573 007555  101  123  123             .ASCII  /ASS/
1574 007560  040  102  000             .ASCII  / B/<00>
1575 007563  000                       .ASCII  <00>
1576 007564  040  124  117  P.AER:     .ASCII  / TO/
1577 007567  040  106  111             .ASCII  / FI/
1578 007572  116  104  000             .ASCII  /ND/<00>
1579 007575  000                       .ASCII  <00>
```

```
1581                                    ;ML4
1582                                    ;
1583
1584 007576  040  116  117  P.AES:  .ASCII  / NO/
1585 007601  124  040  114           .ASCII  /T L/
1586 007604  101  124  103           .ASCII  /ATC/
1587 007607  110  105  104           .ASCII  /HED/
1588 007612  000  000               .ASCII  <00><00>
1589 007614  040  123  111  P.AET:  .ASCII  / SI/
1590 007617  116  107  114           .ASCII  /NGL/
1591 007622  105  040  102           .ASCII  /E B/
1592 007625  111  124  040           .ASCII  /IT /
1593 007630  105  122  122           .ASCII  /ERR/
1594 007633  117  122  000           .ASCII  /OR/<00>
1595 007636  040  115  125  P.AEU:  .ASCII  / MU/
1596 007641  114  124  111           .ASCII  /LTI/
1597 007644  120  114  105           .ASCII  /PLE/
1598 007647  040  102  111           .ASCII  / BI/
1599 007652  124  040  105           .ASCII  /T E/
1600 007655  122  122  117           .ASCII  /RRO/
1601 007660  122  000               .ASCII  /R/<00>
1602 007662  040  040  040  P.AEV:  .ASCII  /   /
1603 007665  040  040  040           .ASCII  /   /
1604 007670  040  040  040           .ASCII  /   /
1605 007673  122  105  107           .ASCII  /REG/
1606 007676  111  123  124           .ASCII  /IST/
1607 007701  105  122  040           .ASCII  /ER /
1608 007704  040  040  104           .ASCII  /  D/
1609 007707  125  115  120           .ASCII  /UMP/
1610 007712  000  000               .ASCII  <00><00>
1611 007714  123  105  122  P.AEW:  .ASCII  /SER/
1612 007717  111  101  114           .ASCII  /IAL/
1613 007722  040  043  000           .ASCII  / #/<00>
1614 007725  000                    .ASCII  <00>
1615 007726  040  115  105  P.AEX:  .ASCII  / ME/
1616 007731  115  040  123           .ASCII  /M S/
1617 007734  111  132  111           .ASCII  /IZI/
1618 007737  116  107  000           .ASCII  /NG/<00>
1619 007742  040  116  117  P.AEY:  .ASCII  / NO/
1620 007745  117  120  000           .ASCII  /OP/<00>
1621 007750  040  104  122  P.AEZ:  .ASCII  / DR/
1622 007753  126  000  000           .ASCII  /V/<00><00>
1623 007756  040  127  122  P.AFA:  .ASCII  / WR/
1624 007761  111  124  105           .ASCII  /ITE/
1625 007764  040  103  110           .ASCII  / CH/
1626 007767  105  103  113           .ASCII  /ECK/
1627 007772  000  000               .ASCII  <00><00>
1628 007774  040  127  122  P.AFB:  .ASCII  / WR/
1629 007777  111  124  105           .ASCII  /ITE/
1630 010002  000  000               .ASCII  <00><00>
1631 010004  040  122  105  P.AFC:  .ASCII  / RE/
1632 010007  101  104  000           .ASCII  /AD/<00>
1633 010012  040  103  114  P.AFD:  .ASCII  / CL/
1634 010015  105  101  122           .ASCII  /EAR/
1635 010020  000  000               .ASCII  <00><00>
```

```
1637                                    ;ML4
1638                                    ;
1639
1640 010022  040  103  117  P.AFE:  .ASCII  / CO/
1641 010025  115  120  040           .ASCII  /MP /
1642 010030  105  122  122           .ASCII  /ERR/
1643 010033  117  122  000           .ASCII  /OR/<00>
1644 010036  040  123  131  P.AFF:  .ASCII  / SY/
1645 010041  123  040  103           .ASCII  /S C/
1646 010044  114  122  000           .ASCII  /LR/<00>
1647 010047  000                     .ASCII  <00>
1648 010050  040  123  105  P.AFG:  .ASCII  / SE/
1649 010053  101  122  103           .ASCII  /ARC/
1650 010056  110  000                .ASCII  /H/<00>
1651 010060  040  122  105  P.AFH:  .ASCII  / RE/
1652 010063  101  104  055           .ASCII  /AD-/
1653 010066  111  116  055           .ASCII  /IN-/
1654 010071  120  122  105           .ASCII  /PRE/
1655 010074  123  105  124           .ASCII  /SET/
1656 010077  000                     .ASCII  <00>
1657 010100  040  111  114  P.AFI:  .ASCII  / IL/
1658 010103  114  105  107           .ASCII  /LEG/
1659 010106  101  114  000           .ASCII  /AL/<00>
1660 010111  000                     .ASCII  <00>
1661 010112  040  101  102  P.AFJ:  .ASCII  / AB/
1662 010115  117  122  124           .ASCII  /ORT/
1663 010120  000  000                .ASCII  <00><00>
1664 010122  040  101  122  P.AFK:  .ASCII  / AR/
1665 010125  122  040  122           .ASCII  /R R/
1666 010130  104  137  127           .ASCII  /D_W/
1667 010133  122  124  000           .ASCII  /RT/<00>
1668 010136  040  107  117  P.AFL:  .ASCII  / GO/
1669 010141  117  104  040           .ASCII  /OD /
1670 010144  102  114  113           .ASCII  /BLK/
1671 010147  000                     .ASCII  <00>
1672 010150  040  122  105  P.AFM:  .ASCII  / RE/
1673 010153  106  122  105           .ASCII  /FRE/
1674 010156  123  110  000           .ASCII  /SH/<00>
1675 010161  000                     .ASCII  <00>
1676 010162  040  101  122  P.AFN:  .ASCII  / AR/
1677 010165  122  101  131           .ASCII  /RAY/
1678 010170  000  000                .ASCII  <00><00>
1679 010172  040  122  101  P.AFO:  .ASCII  / RA/
1680 010175  115  055  102           .ASCII  /M-B/
1681 010200  125  123  000           .ASCII  /US/<00>
1682 010203  000                     .ASCII  <00>
1683 010204  040  117  126  P.AFP:  .ASCII  / OV/
1684 010207  105  122  106           .ASCII  /ERF/
1685 010212  114  117  127           .ASCII  /LOW/
1686 010215  000                     .ASCII  <00>
1687 010216  040  103  110  P.AFQ:  .ASCII  / CH/
1688 010221  113  137  123           .ASCII  /K_S/
1689 010224  125  115  000           .ASCII  /UM/<00>
1690 010227  000                     .ASCII  <00>
1691 010230  040  114  101  P.AFR:  .ASCII  / LA/
```

```
1693                                    ;ML4
1694                                    ;
1695
1696 010233    123     124     040             .ASCII  /ST /
1697 010236    102     114     113             .ASCII  /BLK/
1698 010241    000                             .ASCII  <00>
1699 010242    040     111     116     P.AFS:  .ASCII  / IN/
1700 010245    111     12/     111             .ASCII  /ITI/
1701 010250    101     114     111             .ASCII  /ALI/
1702 010253    132     105     000             .ASCII  /ZE/<00>
1703 010256    040     115     114     P.AFT:  .ASCII  / ML/
1704 010261    103     123     061             .ASCII  /CS1/
1705 010264    000     000                     .ASCII  <00><00>
1706 010266    040     115     114     P.AFU:  .ASCII  / ML/
1707 010271    104     123     000             .ASCII  /DS/<00>
1708 010274    040     115     114     P.AFV:  .ASCII  / ML/
1709 010277    105     122     000             .ASCII  /ER/<00>
1710 010302    040     115     114     P.AFW:  .ASCII  / ML/
1711 010305    115     122     000             .ASCII  /MR/<00>
1712 010310    040     115     114     P.AFX:  .ASCII  / ML/
1713 010313    101     123     000             .ASCII  /AS/<00>
1714 010316    040     115     114     P.AFY:  .ASCII  / ML/
1715 010321    104     101     000             .ASCII  /DA/<00>
1716 010324    040     115     114     P.AFZ:  .ASCII  / ML/
1717 010327    104     124     000             .ASCII  /DT/<00>
1718 010332    040     115     114     P.AGA:  .ASCII  / ML/
1719 010335    120     101     000             .ASCII  /PA/<00>
1720 010340    040     115     114     P.AGB:  .ASCII  / ML/
1721 010343    123     116     000             .ASCII  /SN/<00>
1722 010346    040     115     114     P.AGC:  .ASCII  / ML/
1723 010351    105     061     000             .ASCII  /E1/<00>
1724 010354    040     115     114     P.AGD:  .ASCII  / ML/
1725 010357    105     062     000             .ASCII  /E2/<00>
1726 010362    040     115     114     P.AGE:  .ASCII  / ML/
1727 010365    104     061     000             .ASCII  /D1/<00>
1728 010370    040     115     114     P.AGF:  .ASCII  / ML/
1729 010373    104     062     000             .ASCII  /D2/<00>
1730 010376    040     115     114     P.AGG:  .ASCII  / ML/
1731 010401    105     105     000             .ASCII  /EE/<00>
1732 010404    040     115     114     P.AGH:  .ASCII  / ML/
1733 010407    105     114     000             .ASCII  /EL/<00>
1734 010412    040     115     114     P.AGI:  .ASCII  / ML/
1735 010415    120     104     000             .ASCII  /PD/<00>
1736 010420    040     115     114     P.AGJ:  .ASCII  / ML/
1737 010423    103     123     062             .ASCII  /CS2/
1738 010426    000     000                     .ASCII  <00><00>
1739 010430    040     115     114     P.AGK:  .ASCII  / ML/
1740 010433    127     103     000             .ASCII  /WC/<00>
1741 010436    040     115     114     P.AGL:  .ASCII  / ML/
1742 010441    102     101     000             .ASCII  /BA/<00>
1743 010444    040     115     114     P.AGM:  .ASCII  / ML/
1744 010447    102     101     111             .ASCII  /BAI/
1745 010452    000     000                     .ASCII  <00><00>
1746 010454    040     115     114     P.AGN:  .ASCII  / ML/
1747 010457    103     123     063             .ASCII  /CS3/
```

```
1749                                    ;ML4
1750                                    ;
1751
1752 010462    000     000                     .ASCII  <00><00>
1753 010464    101     123     131     P.AGO:  .ASCII  /ASY/
1754 010467    116     103     110             .ASCII  /NCH/
1755 010472    122     117     116             .ASCII  /RON/
1756 010475    117     125     123             .ASCII  /OUS/
1757 010500    040     106     101             .ASCII  / FA/
1758 010503    111     114     125             .ASCII  /ILU/
1759 010506    122     105     040             .ASCII  /RE /
1760 010511    115     117     104             .ASCII  /MOD/
1761 010514    125     114     105             .ASCII  /ULE/
1762 010517    040     067     063             .ASCII  / 73/
1763 010522    066     061     000             .ASCII  /61/<00>
1764 010525    000                             .ASCII  <00>
1765 010526    123     131     116     P.AGP:  .ASCII  /SYN/
1766 010531    103     110     122             .ASCII  /CHR/
1767 010534    117     116     117             .ASCII  /ONO/
1768 010537    125     123     040             .ASCII  /US /
1769 010542    106     101     111             .ASCII  /FAI/
1770 010545    114     125     122             .ASCII  /LUR/
1771 010550    105     040     115             .ASCII  /E M/
1772 010553    117     104     125             .ASCII  /ODU/
1773 010556    114     105     040             .ASCII  /LE /
1774 010561    067     063     066             .ASCII  /736/
1775 010564    062     040     000             .ASCII  /2 /<00>
1776 010567    000                             .ASCII  <00>
1777 010570    101     122     122     P.AGQ:  .ASCII  /ARR/
1778 010573    101     131     040             .ASCII  /AY /
1779 010576    104     101     124             .ASCII  /DAT/
1780 010601    101     040     106             .ASCII  /A F/
1781 010604    101     111     114             .ASCII  /AIL/
1782 010607    125     122     105             .ASCII  /URE/
1783 010612    040     115     117             .ASCII  / MO/
1784 010615    104     125     114             .ASCII  /DUL/
1785 010620    105     040     067             .ASCII  /E 7/
1786 010623    063     066     063             .ASCII  /363/
1787 010626    040     000                     .ASCII  / /<00>
1788 010630    115     105     115     P.AGR:  .ASCII  /MEM/
1789 010633    117     122     131             .ASCII  /ORY/
1790 010636    040     101     122             .ASCII  / AR/
1791 010641    122     101     131             .ASCII  /RAY/
1792 010644    040     106     101             .ASCII  / FA/
1793 010647    111     114     125             .ASCII  /ILU/
1794 010652    122     105     040             .ASCII  /RE /
1795 010655    115     117     104             .ASCII  /MOD/
1796 010660    125     114     105             .ASCII  /ULE/
1797 010663    040     067     063             .ASCII  / 73/
1798 010666    065     067     000             .ASCII  /57/<00>
1799 010671    000                             .ASCII  <00>
1800 010672    111     116     124     P.AGS:  .ASCII  /INT/
1801 010675    105     122     115             .ASCII  /ERM/
1802 010700    105     104     111             .ASCII  /EDI/
1803 010703    101     124     105             .ASCII  /ATE/
```

```
                                        ;ML4                                    14-May-1981 13:48:54    TOPS
                                        ;                                       14-May-1981 13:05:08    PA:<
                                        .ASCII  / DI/
                                        .ASCII  /AGN/
                                        .ASCII  /OST/
                                        .ASCII  /IC /
                                        .ASCII  /MES/
                                        .ASCII  /SAG/
                                        .ASCII  /E/<00>
                                P.AGT:  .ASCII  /DAT/
                                        .ASCII  /A L/
                                        .ASCII  /ATE/
                                        .ASCII  / ER/
                                        .ASCII  /ROR/
                                        .ASCII  / DU/
                                        .ASCII  /RIN/
                                        .ASCII  /G T/
                                        .ASCII  /RAN/
                                        .ASCII  /SFE/
                                        .ASCII  /R/<00>
                                P.AGU:  .ASCII  /SC /
                                        .ASCII  /BIT/
                                        .ASCII  / SE/
                                        .ASCII  /T D/
                                        .ASCII  /URI/
                                        .ASCII  /NG /
                                        .ASCII  /TRA/
                                        .ASCII  /NSF/
                                        .ASCII  /ER/<00>
                                        .ASCII  <00>
                                P.AGV:  .ASCII  /TRO/
                                        .ASCII  /UBL/
                                        .ASCII  /E S/
                                        .ASCII  /HOO/
                                        .ASCII  /T L/
                                        .ASCII  /OOP/
                                        .ASCII  / ER/
                                        .ASCII  /ROR/
                                        .ASCII  /S/<00>
                                P.AGW:  .ASCII  /RH /
                                        .ASCII  /CON/
                                        .ASCII  /TRO/
                                        .ASCII  /LLE/
                                        .ASCII  /R E/
                                        .ASCII  /RRO/
                                        .ASCII  /RS/<00>
                                        .ASCII  <00>
                                P.AGX:  .ASCII  /DRI/
                                        .ASCII  /VE /
                                        .ASCII  /HUN/
                                        .ASCII  /G A/
                                        .ASCII  /FTE/
                                        .ASCII  /R M/
                                        .ASCII  /ASS/
```

14-May-1981 13:48:54 TOPS
14-May-1981 13:05:08 PA:<

```
                                ;ML4
                                ;
011133      040     102     125         .ASCII  / BU/
011136      123     040     124         .ASCII  /S T/
011141      122     101     116         .ASCII  /RAN/
011144      123     106     105         .ASCII  /SFE/
011147      122     000     000         .ASCII  /R/<00><00>


011152                          NIB.SAVE:
011152                                  .BLKW   3
011160                          HW.OR.TBL:
011160                                  .BLKW   177
011556                          PTBL.PTR:
011556                                  .BLKW   1
011560                          OP.NUM.ARR:
011560                                  .BLKW   1
011562                          ARR.INC:.BLKW   1
011564                          GOOD.BLK:
011564                                  .BLKW   1
011566                          PAR.DIS:.BLKW   1
011570                          CHIP.SIZ:
011570                                  .BLKW   1
011572                          LST.BLK:.BLKW   1
011574                          ARR.16: .BLKW   1
011576                          LST.ARR:.BLKW   1
011600                          IO.BUF: .BLKW   400
012600                          STK.OFF:.BLKW   5
012612                          STACK:  .BLKW   143
013120                          PD.TEMP:.BLKW   1
013122                          W.C.SIZE:
013122                                  .BLKW   1
013124                          RAS.INC:.BLKW   1
013126                          WT.DATA:.BLKW   1
013130                          RD.DATA:.BLKW   1
013132                          DRIVE.TYPE:
013132                                  .BLKW   1
013134                          LST.DUT.REG:
013134                                  .BLKW   1
013136                          REG.INIT.FLG:
013136  000000                          .WORD   0
013140                          A.CAL:  .BLKW   1
013142                          B.CAL:  .BLKW   1
013144                          P.CAL:  .BLKW   1
013146                          A.GEN:  .BLKW   1
013150                          B.GEN:  .BLKW   1
013152                          P.GEN:  .BLKW   1
013154      000                 ML.REG: .BYTE   0
013155      000                         .BYTE   0
013156  004000                          .WORD   400U
013160  173701                          .WORD   -4077
013162  160200                          .WORD   -17600
```

```
1916                            ;ML4
1917                            ;
1918
1919 013164  000                        .BYTE   0
1920 013165  000                        .BYTE   0
1921 013166  000                        .BYTE   0
1922 013167  000                        .BYTE   0
1923 013170  000                        .BYTE   0
1924 013171  000                        .BYTE   0
1925 013172  000                        .BYTE   0
1926 013173  000                        .BYTE   0
1927 013174  000                        .BYTE   0
1928 013175  000                        .BYTE   0
1929 013176  000                        .BYTE   0
1930 013177  000                        .BYTE   0
1931 013200  000                        .BYTE   0
1932 013201  000                        .BYTE   0
1933 013202  000                        .BYTE   0
1934 013203  000                        .BYTE   0
1935 013204  000                        .BYTE   0
1936 013205  000                        .BYTE   0
1937 013206  000                        .BYTE   0
1938 013207  000                        .BYTE   0
1939 013210  000                        .BYTE   0
1940 013211  000                        .BYTE   0
1941 013212  000                        .BYTE   0
1942 013213  000                        .BYTE   0
1943 013214  000                        .BYTE   0
1944 013215  000                        .BYTE   0
1945 013216  000                        .BYTE   0
1946 013217  000                        .BYTE   0
1947 013220  000                        .BYTE   0
1948 013221  000                        .BYTE   0
1949 013222  000                        .BYTE   0
1950 013223  000                        .BYTE   0
1951 013224  000                        .BYTE   0
1952 013225  000                        .BYTE   0
1953 013226  010600                     .WORD   10600
1954 013230  025077                     .WORD   25077
1955 013232  000100                     .WORD   100
1956 013234  000                        .BYTE   0
1957 013235  000                        .BYTE   0
1958 013236  000                        .BYTE   0
1959 013237  000                        .BYTE   0
1960 013240  014620                     .WORD   14620
1961 013242  000                        .BYTE   0
1962 013243  000                        .BYTE   0
1963 013244  000                        .BYTE   0
1964 013245  000                        .BYTE   0
1965 013246  000                        .BYTE   0
1966 013247  000                        .BYTE   0
1967 013250  000                        .BYTE   0
1968 013251  000                        .BYTE   0
1969 013252  177400                     .WORD   -400
1970 013254  000                        .BYTE   0
```

```
                                ;ML4                                    14-May-1981 13:48:54    TOPS
                                ;                                       14-May-1981 13:05:08    PA:<
                                
013255      000                         .BYTE   0
013256      000                         .BYTE   0
013257      000                         .BYTE   0
013260   100000                         .WORD   -100000
013262      000                         .BYTE   0
013263      000                         .BYTE   0
013264      000                         .BYTE   0
013265      000                         .BYTE   0
013266      000                         .BYTE   0
013267      000                         .BYTE   0
013270      000                         .BYTE   0
013271      000                         .BYTE   0
013272      000                         .BYTE   0
013273      000                         .BYTE   0
013274      000                         .BYTE   0
013275      000                         .BYTE   0
013276      000                         .BYTE   0
013277      000                         .BYTE   0
013300   000020                         .WORD   20
013302   177400                         .WORD   -400
013304      000                         .BYTE   0
013305      000                         .BYTE   0
013306   000110                         .WORD   110
013310   177666                         .WORD   -112
013312   000001                         .WORD   1
013314      000                         .BYTE   0
013315      000                         .BYTE   0
013316      000                         .BYTE   0
013317      000                         .BYTE   0
013320      000                         .BYTE   0
013321      000                         .BYTE   0
013322      000                         .BYTE   0
013323      000                         .BYTE   0
013324      000                         .BYTE   0
013325      000                         .BYTE   0
013326      000                         .BYTE   0
013327      000                         .BYTE   0
013330   140300                         .WORD   -37500
013332      000                         .BYTE   0
013333      000                         .BYTE   0
013334      000                         .BYTE   0
013335      000                         .BYTE   0
013336      000                         .BYTE   0
013337      000                         .BYTE   0
013340   100300                         .WORD   -77500
013342      000                         .BYTE   0
013343      000                         .BYTE   0
013344      000                         .BYTE   0
013345      000                         .BYTE   0
013346      000                         .BYTE   0
013347      000                         .BYTE   0
013350      000                         .BYTE   0
```

```
                                ;ML4
                                ;
013351  000                             .BYTE   0
013352  000                             .BYTE   0
013353  000                             .BYTE   0
013354  000                             .BYTE   0
013355  000                             .BYTE   0
013356  000                             .BYTE   0
013357  000                             .BYTE   0
013360  000                             .BYTE   0
013361  000                             .BYTE   0
013362  000                             .BYTE   0
013363  000                             .BYTE   0
013364  000                             .BYTE   0
013365  000                             .BYTE   0
013366  000                             .BYTE   0
013367  000                             .BYTE   0
013370  010000                          .WORD   10000
013372  000                             .BYTE   0
013373  000                             .BYTE   0
013374  000                             .BYTE   0
013375  000                             .BYTE   0
013376  000                             .BYTE   0
013377  000                             .BYTE   0
013400  000                             .BYTE   0
013401  000                             .BYTE   0
013402  000                             .BYTE   0
013403  000                             .BYTE   0
013404  000                             .BYTE   0
013405  000                             .BYTE   0
013406  000                             .BYTE   0
013407  000                             .BYTE   0
013410  000                             .BYTE   0
013411  000                             .BYTE   0
013412  000                             .BYTE   0
013413  000                             .BYTE   0
013414  000                             .BYTE   0
013415  000                             .BYTE   0
013416  000                             .BYTE   0
013417  000                             .BYTE   0
013420  000                             .BYTE   0
013421  000                             .BYTE   0
013422  000                             .BYTE   0
013423  000                             .BYTE   0
013424  000                             .BYTE   0
013425  000                             .BYTE   0
013426  000                             .BYTE   0
013427  000                             .BYTE   0
013430  000                             .BYTE   0
013431  000                             .BYTE   0
013432  000000                          .WORD   0
013434  001001          REM.TBL:        .WORD   1001
013436  004004                          .WORD   4004
013440  020020                          .WORD   20020
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<

                                                .WORD   3003
                                                .WORD   14014
                                                .WORD   21460
                                                .WORD   5005
                                                .WORD   24024
                                                .WORD   23023
                                                .WORD   17017
                                                .WORD   35474
                                                .WORD   24465
                                                .WORD   21021
                                                .WORD   7007
                                                .WORD   34034
                                                .WORD   22463
                                                .WORD   11011
                                                .WORD   5444
                                                .WORD   26026
                                                .WORD   33033
                                                .WORD   16457
                                                .WORD   33472
                                                .WORD   14455
                                                .WORD   23462
                                                .WORD   15015
                                                .WORD   25464
                                                .WORD   25025
                                                .WORD   27027
                                                .WORD   37037
                                                .WORD   36477
                                                .WORD   30471
                                                .BYTE   41
                                                .EVEN
                                        DT.1:   .WORD   64477
                                                .WORD   7700
                                                .BYTE   0
                                                .BYTE   0
                                                .WORD   64477
                                                .WORD   77
                                                .BYTE   0
                                                .BYTE   0
                                                .WORD   -77201
                                                .WORD   17700
                                                .BYTE   77
                                                .BYTE   0
                                                .WORD   -77201
                                                .WORD   0
                                                .BYTE   77
                                                .BYTE   0
                                                .WORD   -77201
                                                .WORD   10077
                                                .BYTE   77
                                                .EVEN
                                        RH.ADD: .BLKW   1
                                        RH.TYP: .BLKW   1
```

```
2140                            ;ML4
2141                            ;
2142
2143 013576                     RH.VEC: .BLKW   1
2144 013600                     ML.LUN: .BLKW   1
2145 013602                     ML.DUT: .BLKW   1
2146
2147
2148          100000            BIT15==             -100000
2149          040000            BIT14==             40000
2150          020000            BIT13==             20000
2151          010000            BIT12==             10000
2152          004000            BIT11==             4000
2153          002000            BIT10==             2000
2154          001000            BIT09==             1000
2155          000400            BIT08==             400
2156          000200            BIT07==             200
2157          000100            BIT06==             100
2158          000040            BIT05==             40
2159          000020            BIT04==             20
2160          000010            BIT03==             10
2161          000004            BIT02==             4
2162          000002            BIT01==             2
2163          000001            BIT00==             1
2164          001000            BIT9==              1000
2165          000400            BIT8==              400
2166          000200            BIT7==              200
2167          000100            BIT6==              100
2168          000040            BIT5==              40
2169          000020            BIT4==              20
2170          000010            BIT3==              10
2171          000004            BIT2==              4
2172          000002            BIT1==              2
2173          000001            BIT0==              1
2174          000040            EF.START==          40
2175          000037            EF.RESTART==        37
2176          000036            EF.CONTINUE==       36
2177          000035            EF.NEW==            35
2178          000034            EF.PWR==            34
2179          000340            PRI07==             340
2180          000300            PRI06==             300
2181          000240            PRI05==             240
2182          000200            PRI04==             200
2183          000140            PRI03==             140
2184          000100            PRI02==             100
2185          000040            PRI01==             40
2186          000000            PRI00==             0
2187          000004            EVL==               4
2188          000010            LOT==               10
2189          000020            A^R==               20
2190          000040            IDU==               40
2191          000100            ISR==               100
2192          000200            UAM==               200
2193          000400            BOE==               400
2194          001000            PNT==               1000
```

```
2196                            ;ML4
2197                            ;
2198
2199    002000                  PRI==           2000
2200    004000                  IXE==           4000
2201    010000                  IBE==           10000
2202    020000                  IER==           20000
2203    040000                  LOE==           40000
2204    100000                  HOE==           -100000
2205    004222                  FMT.1=          P.AAA
2206    004266                  FMT.2=          P.AAB
2207    004354                  FMT.3=          P.AAC
2208    004400                  FMT.4=          P.AAD
2209    004430                  FMT.5=          P.AAE
2210    004532                  FMT.6=          P.AAF
2211    004562                  FMT.7=          P.AAG
2212    004610                  FMT.8=          P.AAH
2213    C04644                  FMT.9=          P.AAI
2214    004676                  FMT.10=         P.AAJ
2215    004750                  FMT.11=         P.AAK
2216    005004                  FMT.12=         P.AAL
2217    005034                  FMT.13=         P.AAM
2218    005110                  FMT.14=         P.AAN
2219    005150                  FMT.15=         P.AAO
2220    005216                  FMT.16=         P.AAP
2221    005306                  FMT.17=         P.AAQ
2222    005342                  FMT.18=         P.AAR
2223    005412                  FMT.19=         P.AAS
2224    005462                  FMT.20=         P.AAT
2225    005534                  FMT.21=         P.AAU
2226    005606                  FMT.22=         P.AAV
2227    005652                  FMT.23=         P.AAW
2228    005722                  FMT.24=         P.AAX
2229    005746                  FMT.25=         P.AAY
2230    005764                  FMT.26=         P.AAZ
2231    006016                  ONE.FMT=        P.ABA
2232    006024                  TWO.FMT=        P.ABB
2233    006034                  THR.FMT=        P.ABC
2234    006046                  FOR.FMT=        P.ABD
2235    006062                  FIV.FMT=        P.ABE
2236    006100                  SIX.FMT=        P.ABF
2237    006120                  SEV.FMT=        P.ABG
2238    006142                  EIG.FMT=        P.ABH
2239    006166                  NIN.FMT=        P.ABI
2240    006214                  TEN.FMT=        P.ABJ
2241    006244                  ELV.FMT=        P.ABK
2242    006276                  WRD.1=          P.ABL
2243    006302                  WRD.2=          P.ABM
2244    006314                  WRD.3=          P.ABN
2245    006322                  WRD.4=          P.ABO
2246    006330                  WRD.5=          P.ABP
2247    006336                  WRD.6=          P.ABQ
2248    006344                  WRD.7=          P.ABR
2249    006360                  WRD.8=          P.ABS
2250    006374                  WRD.9=          P.ABT
```

14-May-1981 13:48:54 TOPS
14-May-1981 13:05:08 PA:<

```
2252                            ;ML4
2253                            ;
2254
2255        006406              WRD.10=         P.ABU
2256        006416              WRD.11=         P.ABV
2257        006426              WRD.12=         P.ABW
2258        006436              WRD.13=         P.ABX
2259        006442              WRD.14=         P.ABY
2260        006454              WRD.15=         P.ABZ
2261        006462              WRD.16=         P.ACA
2262        006470              WRD.17=         P.ACB
2263        006502              WRD.18=         P.ACC
2264        006506              WRD.19=         P.ACD
2265        006514              WRD.20=         P.ACE
2266        006522              WRD.21=         P.ACF
2267        006530              WRD.22=         P.ACG
2268        006544              WRD.23=         P.ACH
2269        006552              WRD.24=         P.ACI
2270        006560              WRD.25=         P.ACJ
2271        006574              WRD.26=         P.ACK
2272        006602              WRD.27=         P.ACL
2273        006610              WRD.29=         P.ACM
2274        006624              WRD.30=         P.ACN
2275        006632              WRD.31=         P.ACO
2276        006646              WRD.32=         P.ACP
2277        006654              WRD.33=         P.ACQ
2278        006660              WRD.34=         P.ACR
2279        006666              WRD.35=         P.ACS
2280        006674              WRD.36=         P.ACT
2281        006700              WRD.37=         P.ACU
2282        006710              WRD.38=         P.ACV
2283        006716              WRD.39=         P.ACW
2284        006726              WRD.40=         P.ACX
2285        006732              WRD.41=         P.ACY
2286        006746              WRD.42=         P.ACZ
2287        006754              WRD.43=         P.ADA
2288        006764              WRD.44=         P.ADB
2289        006772              WRD.45=         P.ADC
2290        007004              WRD.46=         P.ADD
2291        007016              WRD.47=         P.ADE
2292        007026              WRD.48=         P.ADF
2293        007036              WRD.49=         P.ADG
2294        007046              WRD.50=         P.ADH
2295        007054              WRD.51=         P.ADI
2296        007066              WRD.52=         P.ADJ
2297        007074              WRD.53=         P.ADK
2298        007104              WRD.54=         P.ADL
2299        007114              WRD.55=         P.ADM
2300        007122              WRD.56=         P.ADN
2301        007130              WRD.57=         P.ADO
2302        007140              WRD.58=         P.ADP
2303        007152              WRD.59=         P.ADQ
2304        007156              WRD.60=         P.ADR
2305        007174              WRD.61=         P.ADS
2306        007210              WRD.62=         P.ADT
```

```
2308                                ;ML4
2309                                ;
2310
2311        007216                  WRD.63=         P.ADU
2312        007224                  WRD.64=         P.ADV
2313        007232                  WRD.65=         P.ADW
2314        007240                  WRD.67=         P.ADX
2315        007246                  WRD.68=         P.ADY
2316        007252                  WRD.69=         P.ADZ
2317        007260                  WRD.70=         P.AEA
2318        007272                  WRD.71=         P.AEB
2319        007302                  WRD.72=         P.AEC
2320        007310                  WRD.73=         P.AED
2321        007316                  WRD.74=         P.AEE
2322        007330                  WRD.75=         P.AEF
2323        007340                  WRD.76=         P.AEG
2324        007346                  WRD.77=         P.AEH
2325        007366                  PHR.1=          P.AEI
2326        007404                  PHR.2=          P.AEJ
2327        007422                  PHR.3=          P.AEK
2328        007454                  PHR.4=          P.AEL
2329        007472                  PHR.5=          P.AEM
2330        007504                  PHR.6=          P.AEN
2331        007516                  PHR.7=          P.AEO
2332        007540                  PHR.8=          P.AEP
2333        007552                  PHR.9=          P.AEQ
2334        007564                  PHR.10=         P.AER
2335        007576                  PHR.11=         P.AES
2336        007614                  PHR.12=         P.AET
2337        007636                  PHR.13=         P.AEU
2338        007662                  PHR.14=         P.AEV
2339        007714                  PHR.15=         P.AEW
2340        007726                  FNC.1=          P.AEX
2341        007742                  FNC.2=          P.AEY
2342        007750                  FNC.3=          P.AEZ
2343        007756                  FNC.4=          P.AFA
2344        007774                  FNC.5=          P.AFB
2345        010004                  FNC.6=          P.AFC
2346        010012                  FNC.7=          P.AFD
2347        010022                  FNC.8=          P.AFE
2348        010036                  FNC.9=          P.AFF
2349        010050                  FNC.10=         P.AFG
2350        010060                  FNC.11=         P.AFH
2351        010100                  FNC.12=         P.AFI
2352        010112                  FNC.13=         P.AFJ
2353        010122                  FNC.14=         P.AFK
2354        010136                  FNC.15=         P.AFL
2355        010150                  FNC.16=         P.AFM
2356        010162                  FNC.17=         P.AFN
2357        010172                  FNC.18=         P.AFO
2358        010204                  FNC.19=         P.AFP
2359        010216                  FNC.21=         P.AFQ
2360        010230                  FNC.22=         P.AFR
2361        010242                  FNC.23=         P.AFS
2362        010256                  REG.1=          P.AFT
```

```
2364                                    ;ML4
2365                                    ;
2366
2367            010266                  REG.2=          P.AFU
2368            010274                  REG.3=          P.AFV
2369            010302                  REG.4=          P.AFW
2370            010310                  REG.5=          P.AFX
2371            010316                  REG.6=          P.AFY
2372            010324                  REG.7=          P.AFZ
2373            010332                  REG.8=          P.AGA
2374            010340                  REG.9=          P.AGB
2375            010346                  REG.10=         P.AGC
2376            010354                  REG.11=         P.AGD
2377            010362                  REG.12=         P.AGE
2378            010370                  REG.13=         P.AGF
2379            010376                  REG.14=         P.AGG
2380            010404                  REG.15=         P.AGH
2381            010412                  REG.16=         P.AGI
2382            010420                  REG.17=         P.AGJ
2383            010430                  REG.18=         P.AGK
2384            010436                  REG.19=         P.AGL
2385            010444                  REG.20=         P.AGM
2386            010454                  REG.21=         P.AGN
2387            010464                  ASYNC=          P.AGO
2388            010526                  SYNC=           P.AGP
2389            010570                  ARR.DAT=        P.AGQ
2390            010630                  MEM.ARR=        P.AGR
2391            010672                  INTER=          P.AGS
2392            010732                  DATA.LATE=      P.AGT
2393            010772                  SC.SET=         P.AGU
2394            011026                  TRBLE.LOOP=     P.AGV
2395            011060                  RH.ERROR=       P.AGW
2396            011106                  TIME.OUT=       P.AGX
2397            011152                  D1.TEMP=        NIB.SAVE
2398            011154                  D2.TEMP=        NIB.SAVE+2
2399            011156                  E2.TEMP=        NIB.SAVE+4
2400
2401
2402                                            .SBTTL  LOAD.STACK
2403
2407 013604                             LOAD.STACK:
2408 013604  004167  170250                       JSR     R1,$SAVE3               ;                               2223
2409 013610  016601  000012                       MOV     12(SP),R1               ; NIB.PTR,*                     2278
2410 013614  012702  012600                       MOV     #STK.OFF,R2
2411 013620  060102                               ADD     R1,R2
2412 013622  016601  000014                       MOV     14(SP),R1               ; STK.PTR,*
2413 013626  016600  000012                       MOV     12(SP),R0               ; NIB.PTR,*                     2274
2414 013632  006300                               ASL     R0
2415 013634  066007  013640                       ADD     1$(R0),PC
2416 013640  000024                     1$:       .WORD   2$-1$
2417 013642  000042                               .WORD   3$-1$
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                .WORD   4$-1$
                                .WORD   5$-1$
                                .WORD   6$-1$
                                .WORD   7$-1$
                                .WORD   9$-1$
                                .WORD   10$-1$
                                .WORD   12$-1$
                                .WORD   15$-1$
                        2$:     CLR     R3                              ;                       2278
                                BISB    (R2),R3
                                MOV     R1,R0
                                SUB     R3,R0
                                MOV     NIB.SAVE,R3
                                BR      14$
                        3$:     CLR     R3                              ;                       2282
                                BISB    (R2),R3
                                MOV     R1,R0
                                SUB     R3,R0
                                MOV     NIB.SAVE,R3
                                BR      8$
                        4$:     CLR     R3                              ;                       2286
                                BISB    (R2),R3
                                MOV     R1,R0
                                SUB     R3,R0
                                MOV     NIB.SAVE,R3
                                BR      13$
                        5$:     CLR     R3                              ;                       2290
                                BISB    (R2),R3
                                MOV     R1,R0
                                SUB     R3,R0
                                MOV     NIB.SAVE,R3
                                BR      11$
                        6$:     CLR     R3                              ;                       2294
                                BISB    (R2),R3
                                MOV     R1,R0
                                SUB     R3,R0
                                MOV     NIB.SAVE+2,R3
                                BR      14$
                        7$:     CLR     R3                              ;                       2298
                                BISB    (R2),R3
                                MOV     R1,R0
                                SUB     R3,R0
                                MOV     NIB.SAVE+2,R3
                        8$:     ASR     R3
                                ASR     R3
                                ASR     R3
                                ASR     R3
                                BR      14$
                        9$:     CLR     R3                              ;                       2302
                                BISB    (R2),R3
                                MOV     R1,R0
                                SUB     R3,R0
```

```
2475                                    ;ML4                                                    14-May-1981 13:48:54    TOPS
2476                                    ;                                                       14-May-1981 13:05:08    PA:<
2477
2478 014030  016703  175120                      MOV     NIB.SAVE+2,R3
2479 014034  000421                              BR      13$
2480 014036  005003                      10$:    CLR     R3                              ;                               2306
2481 014040  151203                              BISB    (R2),R3
2482 014042  010100                              MOV     R1,R0
2483 014044  160300                              SUB     R3,R0
2484 014046  016703  175102                      MOV     NIB.SAVE+2,R3
2485 014052  006203                      11$:    ASR     R3
2486 014054  006203                              ASR     R3
2487 014056  006203                              ASR     R3
2488 014060  006203                              ASR     R3
2489 014062  000406                              BR      13$
2490 014064  005003                      12$:    CLR     R3                              ;                               2310
2491 014066  151203                              BISB    (R2),R3
2492 014070  010100                              MOV     R1,R0
2493 014072  160300                              SUB     R3,R0
2494 014074  016703  175056                      MOV     NIB.SAVE+4,R3
2495 014100  000303                      13$:    SWAB    R3
2496 014102  042703  177760              14$:    BIC     #177760,R3
2497 014106  105060  012612                      CLRB    STACK(R0)
2498 014112  150360  012612                      BISB    R3,STACK(R0)
2499 014116  000207                              RTS     PC                              ;                               2274
2500 014120  005003                      15$:    CLR     R3                              ;                               2314
2501 014122  151203                              BISB    (R2),R3
2502 014124  160301                              SUB     R3,R1
2503 014126  016703  175024                      MOV     NIB.SAVE+4,R3
2504 014132  006203                              ASR     R3
2505 014134  006203                              ASR     R3
2506 014136  006203                              ASR     R3
2507 014140  006203                              ASR     R3
2508 014142  000303                              SWAB    R3
2509 014144  042703  177770                      BIC     #177770,R3
2510 014150  105061  012612                      CLRB    STACK(R1)
2511 014154  150361  012612                      BISB    R3,STACK(R1)
2512 014160  000207                              RTS     PC                              ;                               2223
2513
2514                                    ; Routine Size:  119 words
2515                                    ; Maximum stack depth per invocation:  4 words
2520
2521
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (6)

;          2319  routine DAT_DM_XFER : novalue =
;          2320
;          2321  !++
;          2322  ! FUNCTIONAL DESCRIPTION:
;          2323  !
;          2324  !       A REPEATEDLY CALLED SEQUENCE
;          2325  !       OF ASSIGNMENT EXPRESSIONS TO
;          2326  !       LOAD THE DSA, BUS ADRS  AND WORD
;          2327  !       COUNT REGISTERS WITH APPROPRIATE
;          2328  !       INFORMATION BEFORE A MASS BUS
;          2329  !       TRANSFER CAN COMMENCE
;          2330  !
;          2331  !       LOADS A MASS BUS BLOCK TRANSFER,
;          2332  !       IN DIAGNOSTIC MODE, AT THE GOOD
;          2333  !       BLOCK ADRS.
;          2334  !
;          2335  !--
;          2336
;          2337      begin
;          2338      DAT_DM = ONE;                     !SET DATA DIAG MODE
;          2339      MLDA = .GOOD_BLK;                 !LOAD DSA REG WITH THE GOOD BLOCK ADRS
;          2340      MLBA = IO_BUF;                    !LOAD BUS ADRS REG WITH THE IO_BUF ADRS
;          2341      MLWC = not 255;                   !LOAD WORD COUNT REG WITH COMPLIMENT 256
;          2342      end;
;
;                                    .SBTTL  DAT.DM.XFER
;014162                         DAT.DM.XFER:
;014162  152777  000010  177104          BISB    #10,@ML.REG+120         ;                2338
;014170  016777  175370  177006          MOV     GOOD.BLK,@ML.REG+30     ;                2339
;014176  012777  011600  176770          MOV     #IO.BUF,@ML.REG+20      ;                2340
;014204  012777  177400  176752          MOV     #-400,@ML.REG+10        ;                2341
;014212  000207                          RTS     PC                      ;                2319
;
;                                    ; Routine Size:  13 words
;                                    ; Maximum stack depth per invocation:  0 words
;
;
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (7)

                2343  routine STRIPPER (WRD_CNT, NIB_CNT) : novalue =
                2344      begin
                2345
                2346  !++
                2347  ! FUNCTIONAL DESCRIPTION:
                2348  !       STRIPPER RUNS IN DATA DIAG MODE AND STRIPS OUT
                2349  !       A VARIABLE NUMBER OF GOOD NIBBLES FROM THE
                2350  !       GOOD BLOCK AND STORES THEM INTO A CONTIGIOUS
                2351  !       STACK FOR SEQUENTIAL ACCESSING
                2352  !
                2353  ! FORMAL PARAMETERS:
                2354  !       WRD_CNT
                2355  !       TELLS STRIPPER HOW MANY WORDS IN THE GOOD
                2356  !       BLOCK TO READ OUT
                2357  !
                2358  !       NIB_CNT
                2359  !       TELLS STRIPPER HOW MANY NIBBLES TO
                2360  !       STRIP OUT OF EACH WORD.
                2361  !
                2362  ! IMPLICIT INPUTS:
                2363  !--
                2364
                2365      local
                2366          STK_PTR;                                !STACK POINTER
                2367
                2368      incr CNT from 0 to .NIB_CNT do              !CLEAR OUT THE STACK OFFSETS
                2369          STK_OFF [.CNT] = ZEROES;
                2370
                2371      CLR_MBUS;
                2372      STK_PTR = -1;                               !RESET THE STACK POINTER
                2373      DAT_DM_XFER ();                             !SET UP A DATA XFER AT THE GOOD BLOCK
                2374      MLCS1 = read;                               !DO A READ XFER
                2375      DELAY (ONE_US);                             !ALLOW PROM DATA TO GET INTO THE MLPD REG
                2376
                2377      incr CNT from 0 to .WRD_CNT do              !LOAD THE STACK WITH ALL GOOD NIBBLE DATA
                2378          begin
                2379          PD_TEMP = .MLPD;                        !GET THE PROM DATA
                2380          DAT_CLK = ONE;                          !CLOCK OUT THE DATA WORD
                2381          DELAY (ONE_US);                         !
                2382          RD_LNG_WRD;                             !READ DATA DIAG REGS INTO NIBBLE SAVE
                2383
                2384          incr NIB_PTR from 0 to .NIB_CNT do      !STRIP OUT X NUMBER OF NIBBLES
                2385              begin
                2386              STK_PTR = .STK_PTR + 1;             !INCREMENT THE STACK POINTER
                2387
                2388              if .PD_TEMP [.NIB_PTR] IS_SET       !SEE IF THIS A GOOD NIBBLE
                2389              then
                2390                  STK_OFF [.NIB_PTR] = (.STK_OFF [.NIB_PTR]) + (.NIB_CNT + 1)
                2391                                                  !THEN INCREMENT IT'S STACK OFFSET
                2392              else
                2393                  LOAD_STACK (.STK_PTR, .NIB_PTR);        !ELSE LOAD THE NIBBLE IN THE STACK
                2394
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (7)
```

```
                                2395                end;
                                2396
                                2397            end;
                                2398
                                2399        CLR_MBUS;
                                2400        end;
                                2641
                                2642                        .GLOBL  L$DLY
                                2643
                                2644
                                2645                        .SBTTL  STRIPPER
2649 014214                             STRIPPER:
2650 014214  004167  167676                    JSR     R1,$SAVE5           ;                                   2343
2651 014220  005746                            TST     -(SP)
2652 014222  016604  000020                    MOV     20(SP),R4           ; NIB.CNT,*                         2368
2653 014226  005001                            CLR     R1                  ; CNT
2654 014230  000403                            BR      2$
2655 014232  105061  012600            1$:     CLRB    STK.OFF(R1)         ; *(CNT)                            2369
2656 014236  005201                            INC     R1                  ; CNT                               2368
2657 014240  020104                    2$:     CMP     R1,R4               ; CNT,*
2658 014242  003773                            BLE     1$
2659 014244  152777  000040  176742            BISB    #40,@ML.REG+40      ;                                   2369
2660 014252  016705  177324                    MOV     ML.DUT,R5
2661 014256  042705  177770                    BIC     #177770,R5
2662 014262  142777  000007  176724            BICB    #7,@ML.REG+40
2663 014270  150577  176720                    BISB    R5,@ML.REG+40
2664 014274  012705  177777                    MOV     #-1,R5              ; *,STK.PTR                         2372
2665 014300  004767  177656                    JSR     PC,DAT.DM.XFER      ;                                   2373
2666 014304  012777  000071  176642            MOV     #71,@ML.REG         ;                                   2374
2667 014312  012701  000001                    MOV     #1,R1               ; *,$$TMP2                          2375
2668 014316  001410                    3$:     BEQ     6$
2669 014320  016702  165572                    MOV     L$DLY,R2            ; *,$$TMP1
2670 014324  001403                            BEQ     5$
2671 014326  005016                    4$:     CLR     (SP)                ; $$TMP
2672 014330  005302                            DEC     R2                  ; $$TMP1
2673 014332  001375                            BNE     4$
2674 014334  005301                    5$:     DEC     R1                  ; $$TMP2
2675 014336  000767                            BR      3$
2676 014340  005003                    6$:     CLR     R3                  ; CNT                               2377
2677 014342  000502                            BR      16$
2678 014344  017767  177034  176546    7$:     MOV     @ML.REG+230,PD.TEMP ;                                   2379
2679 014352  152777  000020  176714            BISB    #20,@ML.REG+120     ;                                   2380
2680 014360  012701  000001                    MOV     #1,R1               ; *,$$TMP2                          2381
2681 014364  001410                    8$:     BEQ     11$
2682 014366  016702  165524                    MOV     L$DLY,R2            ; *,$$TMP1
2683 014372  001403                            BEQ     10$
2684                                    ;ML4                                        14-May-1981 13:48:54    TOPS
2685                                    ;                                           14-May-1981 13:05:08    PA:<
2686
2687 014374  005016                    9$:     CLR     (SP)                ; $$TMP
2688 014376  005302                            DEC     R2                  ; $$TMP1
2689 014400  001375                            BNE     9$
2690 014402  005301                    10$:    DEC     R1                  ; $$TMP2
2691 014404  000767                            BR      8$
2692 014406  017767  176732  174536    11$:    MOV     @ML.REG+170,D1.TEMP
2693 014414  017767  176734  174532            MOV     @ML.REG+200,D2.TEMP
```

```
2694 014422  017767  176706  174526          MOV     @ML.REG+160,E2.TEMP
2695 014430  005002                          CLR     R2                      ; NIB.PTR                       2384
2696 014432  000443                          BR      15$
2697 014434  005205                  12$:    INC     R5                      ; STK.PTR                       2386
2698 014436  010201                          MOV     R2,R1                   ; NIB.PTR,*                     2388
2699 014440  006201                          ASR     R1
2700 014442  006201                          ASR     R1
2701 014444  006201                          ASR     R1
2702 014446  062701  013120                  ADD     #PD.TEMP,R1
2703 014452  010146                          MOV     R1,-(SP)
2704 014454  010246                          MOV     R2,-(SP)                ; NIB.PTR,*
2705 014456  042716  177770                  BIC     #177770,(SP)
2706 014462  012746  000001                  MOV     #1,-(SP)
2707 014466  005046                          CLR     -(SP)
2708 014470  004767  166444                  JSR     PC,BL$GT2
2709 014474  062706  000010                  ADD     #10,SP
2710 014500  005300                          DEC     R0
2711 014502  001011                          BNE     13$
2712 014504  005001                          CLR     R1                      ;                               2390
2713 014506  156201  012600                  BISB    STK.OFF(R2),R1          ; *(NIB.PTR),*
2714 014512  060401                          ADD     R4,R1
2715 014514  010100                          MOV     R1,R0
2716 014516  005200                          INC     R0
2717 014520  110062  012600                  MOVB    R0,STK.OFF(R2)          ; *,*(NIB.PTR)
2718 014524  000405                          BR      14$                     ;                               2388
2719 014526  010546                  13$:    MOV     R5,-(SP)                ; STK.PTR,*                     2393
2720 014530  010246                          MOV     R2,-(SP)                ; NIB.PTR,*
2721 014532  004767  177046                  JSR     PC,LOAD.STACK
2722 014536  022626                          CMP     (SP)+,(SP)+
2723 014540  005202                  14$:    INC     R2                      ; NIB.PTR                       2384
2724 014542  020204                  15$:    CMP     R2,R4                   ; NIB.PTR,*
2725 014544  003733                          BLE     12$
2726 014546  005203                          INC     R3                      ; CNT                           2377
2727 014550  020366  000022          16$:    CMP     R3,22(SP)               ; CNT,WRD.CNT
2728 014554  003673                          BLE     7$
2729 014556  152777  000040  176430          BISB    #40,@ML.REG+40          ;                               2397
2730 014564  016705  177012                  MOV     ML.DUT,R5
2731 014570  042705  177770                  BIC     #177770,R5
2732 014574  142777  000007  176412          BICB    #7,@ML.REG+40
2733 014602  150577  176406                  BISB    R5,@ML.REG+40
2734 014606  005726                          TST     (SP)+                   ;                               2343
2735 014610  000207                          RTS     PC
2736
2737                                 ; Routine Size:  127 words
2738                                 ; Maximum stack depth per invocation:  11 words
2739                                 ;ML4                                    14-May-1981 13:48:54    TOPS
2740                                 ;                                       14-May-1981 13:05:08    PA:<
2741
2746
2747
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (8)

;          2401  routine CAL_CRC : novalue =
;          2402      begin
;          2403
;          2404  !++
;          2405  ! FUNCTIONAL DESCRIPTION:
;          2406  !       CALCULATE THE CRC CODE FOR THE GOOD BLOCKS
;          2407  !       FIRST CRC GROUP (60 NIBBLES) BY STRIPPING
;          2408  !       OUT GOOD NIBBLES AND MODULO 2 ADDING EACH
;          2409  !       BITS WEIGHT RESULTING IN A_CAL,B_CAL,P_CAL.
;          2410  !
;          2411  ! IMPLICIT INPUTS:
;          2412  !       P_CAL,A_CAL,B_CAL
;          2413  !       GOLBAL STORAGE LOCATION WHERE CALCULATED
;          2414  !       CRC CODES ARE STORED AND EXAMINED FROM
;          2415  !
;          2416  !       REM_TBL
;          2417  !       TABLE CONTAINING PRECALCULATED BIT POSITION
;          2418  !       WEIGHTS USED IN THE MODULO 2 CALCULATION OF THE
;          2419  !       CRC CODE
;          2420  !--
;          2421
;          2422      local
;          2423          CHANNEL,                          !CHANNEL POINTS TO THE BITS IN A WORD
;          2424          STK_PTR,                          !STACK POINTER
;          2425          NIB_SAV : bitvector [4],          !NIBBLE SAVE LOCATION
;          2426          ALOG,                             !INDEX INTO REMAINDER TABLE
;          2427          BLOG;                             !INDEX INTO REMAINDER TABLE
;          2428
;          2429      STRIPPER (12, 9);                     !STRIP OUT ALL 10 NIBBLE FROM 13 ARRAY WORDS
;          2430      STK_PTR = -1;                         !RESET STACK POINTER
;          2431
;          2432      incr PLOG from 0 to 5 do              !PLOG POINTS TO ONE CRC GROUP '6 WORDS'
;          2433          begin
;          2434          CHANNEL = -1;                     !RESET CHANNEL FOR EACH WORD
;          2435
;          2436          incr NIB_CNT from 0 to 8 do       !READ DATA NIBBLE 0-8 AND CALCULATE CRC
;          2437              begin
;          2438              STK_PTR = .STK_PTR + 1;       !INCREMENT THE STACK POINTER
;          2439              NIB_SAV = .stack [.STK_PTR];  !LOAD NIV_SAV WITH A STACK NIBBLE
;          2440
;          2441              incr BIT_TST from 0 to 3 do   !TEST THE BIT FOR BEING SET
;          2442                  begin
;          2443                  CHANNEL = .CHANNEL + 1;   !CHANNEL POINTS TO THIS BIT
;          2444
;          2445                  if .NIB_SAV [.BIT_TST] IS_SET   !SEE IF THIS BIT IS SET
;          2446                  then
;          2447                      begin                 !IF SET THEN MOD 2 ADD THE REMAINDER
;          2448                      ALOG = .PLOG + .CHANNEL;    !CALCULATE THE ALOG
;          2449                      BLOG = .PLOG + .CHANNEL*2;  !CALCULATE THE BLOG
;          2450
;          2451                      while .ALOG geq 63 do       !REDUCE ALOG UNTIL < 64
;          2452                          ALOG = .ALOG - 63;
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (8)
```

```
                        while .BLOG geq 63 do       !REDUCE BLOG UNTIL <64
                            BLOG = .BLOG - 63;

                        P_CAL = (.P_CAL) xor (.REM_TBL [.PLOG]);     !MOD 2 P_CAL WITH REM_TBL
                        A_CAL = (.A_CAL) xor (.REM_TBL [.ALOG]);     !MOD 2 A_CAL WITH REM_TBL
                        B_CAL = (.B_CAL) xor (.REM_TBL [.BLOG]);     !MOD 2 B_CAL WITH REM_TBL
                        end;

                    end;

                end;

            STK_PTR = .STK_PTR + 1;                 !SKIP OVER THE CRC NIB NT THE STACK
            end;

        end;

                                .SBTTL  CAL.CRC
014612  004167  167300         CAL.CRC:JSR     R1,$SAVE5                       ;                                       2401
014616  162706  000012                 SUB     #12,SP
014622  012746  000014                 MOV     #14,-(SP)                       ;                                       2429
014626  012746  000011                 MOV     #11,-(SP)
014632  004767  177356                 JSR     PC,STRIPPER
014636  012766  177777  000006         MOV     #-1,6(SP)                       ; *,STK.PTR                             2430
014644  005066  000004                 CLR     4(SP)                           ; PLOG                                  2432
014650  012766  177777  000010  1$:    MOV     #-1,10(SP)                      ; *,CHANNEL                             2434
014656  005066  000012                 CLR     12(SP)                          ; NIB.CNT                               2436
014662  005266  000006         2$:     INC     6(SP)                           ; STK.PTR                               2438
014666  016605  000006                 MOV     6(SP),R5                        ; STK.PTR,*                             2439
014672  116566  012612  000014         MOVB    STACK(R5),14(SP)                ; *,NIB.SAV
014700  105066  000015                 CLRB    15(SP)                          ; NIB.SAV
014704  005003                         CLR     R3                              ; BIT.TST                               2441
014706  005266  000010         3$:     INC     10(SP)                          ; CHANNEL                               2443
014712  010305                         MOV     R3,R5                           ; BIT.TST,*                             2445
014714  006205                         ASR     R5
014716  006205                         ASR     R5
014720  006205                         ASR     R5
014722  012704  000014                 MOV     #14,R4
014726  060604                         ADD     SP,R4                           ; NIB.SAV,*
014730  060405                         ADD     R4,R5
014732  010546                         MOV     R5,-(SP)
014734  010346                         MOV     R3,-(SP)                        ; BIT.TST,*
014736  042716  177770                 BIC     #177770,(SP)
014742  012746  000001                 MOV     #1,-(SP)
014746  005046                         CLR     -(SP)
                        ;ML4                                                                    14-May-1981 13:48:54    TOPS
                        ;                                                                       14-May-1981 13:05:08    PA:<

014750  004767  166164                 JSR     PC,BL$GT2
014754  062706  000010                 ADD     #10,SP
014760  005300                         DEC     R0
014762  001066                         BNE     7$
014764  016601  000004                 MOV     4(SP),R1                        ; PLOG,ALOG                             2448
014770  066601  000010                 ADD     10(SP),R1                       ; CHANNEL,ALOG
014774  016605  000010                 MOV     10(SP),R5                       ; CHANNEL,*                             2449
```

```
2869 015000  006305                    ASL     R5
2870 015002  066605  000004            ADD     4(SP),R5                ; PLOG,*
2871 015006  010502                    MOV     R5,R2                   ; *,BLOG
2872 015010  020127  000077      4$:   CMP     R1,#77                  ; ALOG,*                2451
2873 015014  002403                    BLT     5$
2874 015016  162701  000077            SUB     #77,R1                  ; *,ALOG                2452
2875 015022  000772                    BR      4$                      ;                       2451
2876 015024  020227  000077      5$:   CMP     R2,#77                  ; BLOG,*                2454
2877 015030  002403                    BLT     6$
2878 015032  162702  000077            SUB     #77,R2                  ; *,BLOG                2455
2879 015036  000772                    BR      5$                      ;                       2454
2880 015040  005005              6$:   CLR     R5                      ;                       2457
2881 015042  016604  000004            MOV     4(SP),R4                ; PLOG,*
2882 015046  156405  013434            BISB    REM.TBL(R4),R5
2883 015052  010546                    MOV     R5,-(SP)
2884 015054  046716  176064            BIC     P.CAL,(SP)
2885 015060  040567  176060            BIC     R5,P.CAL
2886 015064  052667  176054            BIS     (SP)+,P.CAL
2887 015070  005005                    CLR     R5                      ;                       2458
2888 015072  156105  013434            BISB    REM.TBL(R1),R5          ; *(ALOG),*
2889 015076  010504                    MOV     R5,R4
2890 015100  046704  176034            BIC     A.CAL,R4
2891 015104  040567  176030            BIC     R5,A.CAL
2892 015110  050467  176024            BIS     R4,A.CAL
2893 015114  005005                    CLR     R5                      ;                       2459
2894 015116  156205  013434            BISB    REM.TBL(R2),R5          ; *(BLOG),*
2895 015122  010504                    MOV     R5,R4
2896 015124  046704  176012            BIC     B.CAL,R4
2897 015130  040567  176006            BIC     R5,B.CAL
2898 015134  050467  176002            BIS     R4,B.CAL
2899 015140  005203              7$:   INC     R3                      ; BIT.TST               2441
2900 015142  020327  000003            CMP     R3,#3                   ; BIT.TST,*
2901 015146  003657                    BLE     3$
2902 015150  005266  000012            INC     12(SP)                  ; NIB.CNT               2436
2903 015154  026627  000012  000010    CMP     12(SP),#10              ; NIB.CNT,*
2904 015162  003637                    BLE     2$
2905 015164  005266  000006            INC     6(SP)                   ; STK.PTR               2466
2906 015170  005266  000004            INC     4(SP)                   ; PLOG                  2432
2907 015174  026627  000004  000005    CMP     4(SP),#5                ; PLOG,*
2908 015202  003622                    BLE     1$
2909 015204  062706  000016            ADD     #16,SP                  ;                       2401
2910 015210  000207                    RTS     PC
2911
2912                             ; Routine Size:  128 words
2913                             ; Maximum stack depth per invocation:  17 words
2914                             ;ML4                                    14-May-1981 13:48:54    TOPS
2915                             ;                                       14-May-1981 13:05:08    PA:<
2916
2921
2922
```

```
2924 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
2925 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (9)
2926
2927 ;          2470  routine ERR_CHK_CRC =
2928 ;          2471      begin
2929 ;          2472
2930 ;          2473  !++
2931 ;          2474  ! FUNCTIONAL DESCRIPTION:
2932 ;          2475  !       'ERROR CHECK CRC' COMPARES HARDWARE GENERATED CRC
2933 ;          2476  !       CODES TO SOFTWARE CALCULATED CRC CODE FOR ONE
2934 ;          2477  !       CRC GROUP AND RETURNS A ONE ON DETECTION OF ERRORS
2935 ;          2478  !
2936 ;          2479  ! IMPLICIT INPUTS:
2937 ;          2480  !       P_CAL,A_CAL,B_CAL
2938 ;          2481  !       GLOBAL STORAGE LOCATIONS WHERE CALCULATED
2939 ;          2482  !       CRC CODES ARE STORED AND EXAMINED FROM
2940 ;          2483  !
2941 ;          2484  !       P_GEN,A_GEN,B_GEN
2942 ;          2485  !       GLOBAL STROAGE LOCATIONS WHERE HARDWARE GENERATED
2943 ;          2486  !       CRC CODES ARE STORED AND EXAMINED FROM
2944 ;          2487  !--
2945 ;          2488
2946 ;          2489      local
2947 ;          2490          POS,                                  !VARIABLE FOR FIELD SELECTOR
2948 ;          2491          TEMP;                                 !TEMPORARY STORAGE LOCATION
2949 ;          2492
2950 ;          2493      POS = -1;                                 !RESET THE POS
2951 ;          2494
2952 ;          2495      incr STK_PTR from 9 to 59 by 10 do        !CONVERT VERTICAL STACK CRC TO HORIZONTAL REPRESENTATION
2953 ;          2496          begin
2954 ;          2497          POS = .POS + 1;                       !INCREMENT POS
2955 ;          2498          TEMP = .stack [.STK_PTR];             !READ CRC NIB FROM THE STACK
2956 ;          2499          (P_GEN)<.POS, 1> = .TEMP<0, 1>;       !CONVERT 1ST BIT
2957 ;          2500          (A_GEN)<.POS, 1> = not .TEMP<1, 1>;   !CONVERT 2ND BIT
2958 ;          2501          (B_GEN)<.POS, 1> = not .TEMP<2, 1>;   !CONVERT 3ED BIT
2959 ;          2502          end;
2960 ;          2503
2961 ;          2504      if (.P_GEN neq .P_CAL) or (.A_GEN neq .A_CAL) or (.B_GEN neq .B_CAL) then return ONE else return ZERO;
2962 ;          2505
2963 ;          2506                                                !COMPARE GENERATED CRC CODE TO CALCULATED CRC CODE
2964 ;          2507      end;
2968
2969                                      .SBTTL  ERR.CHK.CRC
2973 015212                          ERR.CHK.CRC:
2974 015212  004167  166642                 JSR     R1,$SAVE3                ;                                  2470
2975 015216  012701  177777                 MOV     #-1,R1                   ; *,POS                            2493
2976 015222  012702  000011                 MOV     #11,R2                   ; *,STK.PTR                        2495
2977 015226  005201                  1$:    INC     R1                       ; POS                              2497
2978 015230  005003                         CLR     R3                       ; TEMP                             2498
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                                BISB    STACK(R2),R3            ; *(STK.PTR),TEMP
                                                MOV     #P.GEN,-(SP)            ;                                       2499
                                                MOV     R1,-(SP)                ; POS,*
                                                MOV     #1,-(SP)
                                                MOV     R3,-(SP)                ; TEMP,*
                                                BIC     #177776,(SP)
                                                JSR     PC,BL$PU2
                                                MOV     #A.GEN,(SP)             ;                                       2500
                                                MOV     R1,-(SP)                ; POS,*
                                                MOV     #1,-(SP)
                                                CLR     -(SP)
                                                BIT     #2,R3                   ; *,TEMP
                                                BEQ     2$
                                                INC     (SP)
                                        2$:     COM     (SP)
                                                JSR     PC,BL$PU2
                                                MOV     #B.GEN,(SP)             ;                                       2501
                                                MOV     R1,-(SP)                ; POS,*
                                                MOV     #1,-(SP)
                                                CLR     -(SP)
                                                BIT     #4,R3                   ; *,TEMP
                                                BEQ     3$
                                                INC     (SP)
                                        3$:     COM     (SP)
                                                JSR     PC,BL$PU2
                                                ADD     #24,SP                  ;                                       2496
                                                ADD     #12,R2                  ; *,STK.PTR                             2495
                                                CMP     R2,#73                  ; STK.PTR,*
                                                BLE     1$
                                                CMP     P.GEN,P.CAL             ;                                       2504
                                                BNE     4$
                                                CMP     A.GEN,A.CAL
                                                BNE     4$
                                                CMP     B.GEN,B.CAL
                                                BEQ     5$
                                        4$:     MOV     #1,R1
                                                BR      6$
                                        5$:     CLR     R1
                                        6$:     MOV     R1,R0                   ;                                       2471
                                                RTS     PC                      ;                                       2470

                                        ; Routine Size:  71 words
                                        ; Maximum stack depth per invocation:  14 words
```

```
3033 ;ML4
3034 ;
3035
3036 ;          2508  routine FIND_COMP_BIT (BITS_XFERED) =
3037 ;          2509      begin
3038 ;          2510
3039 ;          2511  !++
3040 ;          2512  ! FUNCTIONAL DESCRIPTION:
3041 ;          2513  !       'FIND COMPLIMENT'  BIT IS CALLED BY THE
3042 ;          2514  !       SYNDROME DECODE TESTS AND INDEXES INTO
3043 ;          2515  !       THE IO_BUF LOOKING FOR COMPLIMENTED
3044 ;          2516  !       BITS.  A COMPLIMENTED BIT IS BY DEF A ONE
3045 ;          2517  !
3046 ;          2518  !       A ONE IS RETURNED IF THE INDEXED BIT IS
3047 ;          2519  !       NOT SET
3048 ;          2520  !
3049 ;          2521  ! FORMAL PARAMETERS:
3050 ;          2522  !       BITS_XFERED
3051 ;          2523  !       REPRESENTS THE BIT BEING TESTED
3052 ;          2524  !       IN THE DRIVE AND FROM THIS THE
3053 ;          2525  !       WORD AND BIT INDEX INTO THE
3054 ;          2526  !       IO_BUF CAN BE CALCULATED
3055 ;          2527  !
3056 ;          2528  ! IMPLICIT INPUTS:
3057 ;          2529  !--
3058 ;          2530
3059 ;          2531      local
3060 ;          2532          COMP_WRD,                       !WORD WHERE COMPLINEMT BIT IS LOCATED
3061 ;          2533          COMP_BIT,                       !BIT THAT IS COMPLINEMTED
3062 ;          2534          BUF_SAV : bitvector [16];       !STORES THE WORD IN WHICH THE BIT IS COMPLIMENTED
3063 ;          2535
3064 ;          2536      COMP_WRD = .BITS_XFERED/16;         !CALCULATE THE COMP WORD
3065 ;          2537      COMP_BIT = .BITS_XFERED mod 16;     !CALCULATE THE COMP BIT
3066 ;          2538      BUF_SAV = .IO_BUF [.COMP_WRD];      !LOAD THE COMP WORD INTO BUF_SAV
3067 ;          2539      IO_BUF [.COMP_WRD] = ZEROES;        !ZERO THE COMP WORD
3068 ;          2540
3069 ;          2541      if .BUF_SAV [.COMP_BIT] IS_NOT_SET then return ZERO else return ONE;
3070 ;          2542
3071 ;          2543                                          !FIND COMPLIMENTED BIT AND RETURN ERROR STATUS
3072 ;          2544      end;
3076
3077                                    .SBTTL  FIND.COMP.BIT
3081 015430                        FIND.COMP.BIT:
3082 015430  004167  166410                JSR     R1,$SAVE2                       ;                               2508
3083 015434  005746                        TST     -(SP)
3084 015436  016646  000012                MOV     12(SP),-(SP)                    ; BITS.XFERED,*                 2536
3085 015442  012746  000020                MOV     #20,-(SP)
3086 015446  004767  166300                JSR     PC,BL$DIV
3087 015452  010001                        MOV     R0,R1                           ; *,COMP.WRD
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                                MOV     16(SP),(SP)             ; BITS.XFERED,*                         2537
                                                MOV     #20,-(SP)
                                                JSR     PC,BL$MOD
                                                MOV     R0,R2                   ; *,COMP.BIT
                                                MOV     R1,R0                   ; COMP.WRD,*                            2538
                                                ASL     R0
                                                MOV     IO.BUF(R0),6(SP)        ; *,BUF.SAV
                                                CLR     IO.BUF(R0)              ;                                       2539
                                                MOV     R2,R0                   ; COMP.BIT,*                            2541
                                                ASR     R0
                                                ASR     R0
                                                ASR     R0
                                                MOV     #6,R1
                                                ADD     SP,R1                   ; BUF.SAV,*
                                                ADD     R1,R0
                                                MOV     R0,-(SP)
                                                MOV     R2,-(SP)                ; COMP.BIT,*
                                                BIC     #177770,(SP)
                                                MOV     #1,-(SP)
                                                CLR     -(SP)
                                                JSR     PC,BL$GT2
                                                ADD     #10,SP
                                                TST     R0
                                                BNE     1$
                                                CLR     R1
                                                BR      2$
                                        1$:     MOV     #1,R1
                                        2$:     MOV     R1,R0                   ;                                       2509
                                                ADD     #10,SP                  ;                                       2508
                                                RTS     PC

                                        ; Routine Size:  53 words
                                        ; Maximum stack depth per invocation:  11 words
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (11)

;          2545  routine FORCE_REM (PLOG, CHAN) : novalue =
;          2546      begin
;          2547
;          2548  !++
;          2549  ! FUNCTIONAL DESCRIPTION:
;          2550  !       VIA ECC_DM AND THE PRECALCULATED MODULO 2
;          2551  !       REMAINDERS STORED IN REM_TBL[], THE
;          2552  !       CRC_A,CRC_B AND PAR_CRC_WRD
;          2553  !       ARE FORCED WITH ERRONEOUS CRC DATA TO
;          2554  !       FORCE PERDICTABLE ECC ERRORS
;          2555  !
;          2556  ! FORMAL PARAMETERS:
;          2557  !       PLOG
;          2558  !       POINTS TO ONE OF THE SIX WORDS OF
;          2559  !       A CRC GROUP
;          2560  !
;          2561  !       CHAN
;          2562  !       POINTS TO ONE OF 36 CHANNELS IN
;          2563  !       A CRC GROUP
;          2564  !
;          2565  ! IMPLICIT INPUTS:
;          2566  !--
;          2567
;          2568      local
;          2569          ALOG,                                   !INDEX INTO REM_TBL
;          2570          BLOG;                                   !INDEX INTO REM_TBL
;          2571
;          2572      ALOG = .PLOG + .CHAN;                       !CALCULATE A_LOG
;          2573      BLOG = .PLOG + .CHAN*2;                     !CALCULATE B_LOG
;          2574
;          2575      while .ALOG geq 63 do                       !REDUCE A_LOG UNTIL < 64
;          2576          ALOG = .ALOG - 63;
;          2577
;          2578      while .BLOG geq 63 do                       !REDUCE B_LOG UNTIL < 64
;          2579          BLOG = .BLOG - 63;
;          2580
;          2581      ECC_DM = ONE;                               !SET ECC DIAG MODE
;          2582      PAR_CRC_WRD = .REM_TBL [.PLOG];             !LOAD PAR_CRC_WRD WITH REM_TBL
;          2583      CRC_A = .REM_TBL [.ALOG];                   !LOAD CRC_A WITH REM_TBL
;          2584      CRC_B = .REM_TBL [.BLOG];                   !LOAD CRC_B WITH REM_TBL
;          2585      end;
;3179
;3180                                        .SBTTL  FORCE.REM
;3184 015602                           FORCE.REM:
;3185 015602  004167  166252                  JSR     R1,$SAVE3                       ;                       2545
;3186 015606  016600  000014                  MOV     14(SP),R0                       ; PLOG,*                2572
```

```
3188                                    ;ML4                                                    14-May-1981 13:48:54    TOPS
3189                                    ;                                                       14-May-1981 13:05:08    PA:<
3190
3191 015612  010002                            MOV     R0,R2                   ; *,ALOG
3192 015614  066602  000012                    ADD     12(SP),R2               ; CHAN,ALOG
3193 015620  016601  000012                    MOV     12(SP),R1               ; CHAN,*                                2573
3194 015624  006301                            ASL     R1
3195 015626  060001                            ADD     R0,R1
3196 015630  020227  000077            1$:     CMP     R2,#77                  ; ALOG,*                                2575
3197 015634  002403                            BLT     2$
3198 015636  162702  000077                    SUB     #77,R2                  ; *,ALOG                                2576
3199 015642  000772                            BR      1$                      ;                                       2575
3200 015644  020127  000077            2$:     CMP     R1,#77                  ; BLOG,*                                2578
3201 015650  002403                            BLT     3$
3202 015652  162701  000077                    SUB     #77,R1                  ; *,BLOG                                2579
3203 015656  000772                            BR      2$                      ;                                       2578
3204 015660  152777  000001  175406    3$:     BISB    #1,@ML.REG+120          ;                                       2581
3205 015666  116003  013434                    MOVB    REM.TBL(R0),R3          ;                                       2582
3206 015672  000303                            SWAB    R3
3207 015674  042703  140377                    BIC     #140377,R3
3208 015700  042777  037400  175416            BIC     #37400,@ML.REG+150
3209 015706  050377  175412                    BIS     R3,@ML.REG+150
3210 015712  116203  013434                    MOVB    REM.TBL(R2),R3          ; *(ALOG),*                             2583
3211 015716  042703  177700                    BIC     #177700,R3
3212 015722  042777  000077  175374            BIC     #77,@ML.REG+150
3213 015730  050377  175370                    BIS     R3,@ML.REG+150
3214 015734  116103  013434                    MOVB    REM.TBL(R1),R3          ; *(BLOG),*                             2584
3215 015740  042703  177700                    BIC     #177700,R3
3216 015744  042777  000077  175362            BIC     #77,@ML.REG+160
3217 015752  050377  175356                    BIS     R3,@ML.REG+160
3218 015756  000207                            RTS     PC                      ;                                       2545
3219
3220                                    ; Routine Size:  55 words
3221                                    ; Maximum stack depth per invocation:  4 words
3226
3227
```

```
3229 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
3230 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (12)
3231
3232 ;          2586  routine FOR_ECC_ERR (SGL, UNC) : novalue =
3233 ;          2587      begin
3234 ;          2588
3235 ;          2589  !++
3236 ;          2590  ! FUNCTIONAL DESCRIPTION:
3237 ;          2591  !       'FORCE ECC ERROR' WHEN CALLED WILL FORCE
3238 ;          2592  !       VIA ECC DIAG MODE, SINGLE BIT ECC ERRORS
3239 ;          2593  !       OR UNCORRECTABLE ECC ERRORS INTO THE
3240 ;          2594  !       ECC DIAGNOSTIC REGISTERS
3241 ;          2595  !
3242 ;          2596  ! FORMAL PARAMETERS:
3243 ;          2597  !       SGL
3244 ;          2598  !       WHEN SET TO A ONE THIS
3245 ;          2599  !       ROUTINE WILL FORCE SINGLE BIT ERRORS
3246 ;          2600  !
3247 ;          2601  !       UNC
3248 ;          2602  !       WHEN SET TO A ONE THIS ROUTINE WILL
3249 ;          2603  !       FORCE UNCORRECTABLE ECC ERRORS
3250 ;          2604  !
3251 ;          2605  ! IMPLICIT INPUTS:
3252 ;          2606  !--
3253 ;          2607
3254 ;          2608      ECC_DM = ONE;                          !SET ECC DIAG MODE
3255 ;          2609
3256 ;          2610      if .SGL                                !IF SGL IS TRUE
3257 ;          2611      then
3258 ;          2612          begin                              !THEN FORCE A SINGLE BIT ERROR
3259 ;          2613          PAR_CRC_WRD = %b'111111';
3260 ;          2614          CRC_A = %b'000000';
3261 ;          2615          CRC_B = %b'000000';
3262 ;          2616          end
3263 ;          2617      else
3264 ;          2618          begin                              !ELSE FORCE A UNCORRECTABLE ERROR
3265 ;          2619          PAR_CRC_WRD = %b'111111';
3266 ;          2620          CRC_A = %b'111111';
3267 ;          2621          CRC_B = %b'000000';
3268 ;          2622          end;
3269 ;          2623
3270 ;          2624      end;
3274
3275                                          .SBTTL  FOR.ECC.ERR
3279 015760                          FOR.ECC.ERR:
3280 015760  152777  000001  175306          BISB    #1,@ML.REG+120          ;                       2608
3281 015766  032766  000001  000004          BIT     #1,4(SP)                ; *,SGL                 2610
3282 015774  001407                          BEQ     1$
3283 015776  052777  037400  175320          BIS     #37400,@ML.REG+150      ;                       2613
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                                                                                                        2614
016004  042777  000077  175312          BIC     #77,@ML.REG+150                 ;               2615
016012  000406                          BR      2$                              ;               2619
016014  052777  037400  175302  1$:     BIS     #37400,@ML.REG+150              ;               2620
016022  052777  000077  175274          BIS     #77,@ML.REG+150                 ;               2621
016030  042777  000077  175276  2$:     BIC     #77,@ML.REG+160                 ;               2586
016036  000207                          RTS     PC                              ;
                                        ; Routine Size:  24 words
                                        ; Maximum stack depth per invocation:  0 words
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (13)

;          2625  routine FIRST_BLK_XFER : novalue =
;          2626
;          2627  !++
;          2628  ! FUNCTIONAL DESCRIPTION:
;          2629  !       A REPEATEDLY CALLED SEQUENCE OF
;          2630  !       ASSIGNMENT EXPRESSION TO LOAD
;          2631  !       THE DSA, BUS ADRS AND WORD COUNT
;          2632  !       REGISTERS WITH APPROPRIATE INFORMATION
;          2633  !       BEFORE MASS BUS TRANSFERS CAN
;          2634  !       COMMENCE.
;          2635  !
;          2636  !       LOADS A MASS_BUS BLOCK XFERR AT
;          2637  !       BLOCK ZERO.
;          2638  !
;          2639  !--
;          2640
;          2641      begin
;          2642      ECC_DIS = ONE;                    !DISABLE ECC CORRECTION
;          2643      MLDA = ZEROES;                    !LOAD THE DSA REG WITH SECTOR ZERO
;          2644      MLBA = IO_BUF;                    !LOAD THE BUS ADDRESS REG WITH IO_BUF ADRS
;          2645      MLWC = not 255;                   !LOAD WORD COUNT REG WITH COMPLIMENT 256
;          2646      end;
;
                                        .SBTTL  FIRST.BLK.XFER
                            FIRST.BLK.XFER:
016040                                  BISB    #2,@ML.REG+120          ;                       2642
016046  005077  175132                  CLR     @ML.REG+30              ;                       2643
016052  012777  011600  175114          MOV     #IO.BUF,@ML.REG+20      ;                       2644
016060  012777  177400  175076          MOV     #-400,@ML.REG+10        ;                       2645
016066  000207                          RTS     PC                      ;                       2625

                            ; Routine Size:  12 words
                            ; Maximum stack depth per invocation:  0 words
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (14)

;          2647  routine GD_BLK_XFER : novalue =
;          2648
;          2649  !++
;          2650  ! FUNCTIONAL DESCRIPTION:
;          2651  !       A REPEATEDLY CALLED SEQUENCE OF
;          2652  !       ASSIGNMENT EXPRESSIONS TO LOAD
;          2653  !       THE DSA, BUS ADRS AND WORD
;          2654  !       COUNT REGISTERS WITH APPROPRIATE
;          2655  !       INFORMATION BEFORE A MASS BUS
;          2656  !       TRANSFERS CAN COMMENCE
;          2657  !
;          2658  !       LOADS A MASS BUS BLOCK XFERR
;          2659  !       AT THE GOOD BLOCK ADRS FOUND
;          2660  !       BY THE READ WRITE ARRAYS WITH
;          2661  !       PROM DATA TEST
;          2662  !
;          2663  !--
;          2664
;          2665      begin
;          2666      ECC_DIS = ONE;                    !DISABLE ERROR CORRECTION
;          2667      MLDA = .GOOD_BLK;                 !LOAD DSA REG WITH THE GOOD BLOCK ADRS
;          2668      MLBA = IO_BUF;                    !LOAD BUS ADRS REG WITH IO_BUF ADRS
;          2669      MLWC = not 255;                   !LOAD WORD COUNT REG WITH COMPLIMENT 256
;          2670      end;
3383
3384                                        .SBTTL  GD.BLK.XFER
3388 016070                           GD.BLK.XFER:
3389 016070  152777  000002  175176          BISB    #2,@ML.REG+120          ;                       2666
3390 016076  016777  173462  175100          MOV     GOOD.BLK,@ML.REG+30     ;                       2667
3391 016104  012777  011600  175062          MOV     #IO.BUF,@ML.REG+20      ;                       2668
3392 016112  012777  177400  175044          MOV     #-400,@ML.REG+10        ;                       2669
3393 016120  000207                          RTS     PC                      ;                       2647
3394
3395                                  ; Routine Size:  13 words
3396                                  ; Maximum stack depth per invocation:  0 words
3401
3402
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (15)

;          2671  routine LAST_BLK_XFER : novalue =
;          2672
;          2673  !++
;          2674  ! FUNCTIONAL DESCRIPTION:
;          2675  !       A REPEATEDLY CALLED SEQUENCE OF
;          2676  !       ASSIGNMENT EXPRESSIONS TO LOAD
;          2677  !       THE DSA, BUS ADRS AND WORD
;          2678  !       COUNT REGISTERS WITH APPROPRIATE
;          2679  !       INFORMATION BEFORE A MBUS
;          2680  !       TRANSFER CAN COMMENCE
;          2681  !
;          2682  !       LOADS A MASS BUS BLOCK
;          2683  !       TRANSFER AT THE LAST BLOCK
;          2684  !       ADDRESS
;          2685  !
;          2686  !--
;          2687
;          2688      begin
;          2689      ECC_DIS = ONE;                    !DISABLE ERROR CORRECTION
;          2690      MLDA = .LST_BLK;                  !LOAD DSA REG WITH THE LAST BLOCK ADRS
;          2691      MLBA = IO_BUF;                    !LOAD BUS ADRS REG WITH THE IO_BUF ADRS
;          2692      MLWC = not 255;                   !LOAD WORD COUNT REG WITH COMPLIMENT 256
;          2693      end;
;
;                                    .SBTTL  LAST.BLK.XFER
;016122                         LAST.BLK.XFER:
;016122  152777  000002  175144          BISB    #2,@ML.REG+120          ;  2689
;016130  016777  173436  175046          MOV     LST.BLK,@ML.REG+30      ;  2690
;016136  012777  011600  175030          MOV     #IO.BUF,@ML.REG+20      ;  2691
;016144  012777  177400  175012          MOV     #-400,@ML.REG+10        ;  2692
;016152  000207                          RTS     PC                      ;  2671
;
;                                    ; Routine Size:  13 words
;                                    ; Maximum stack depth per invocation:  0 words
;
;
```

```
3454 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
3455 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (16)
3456
3457 ;          2694  routine TST_LNG_WRD (NIB_NUM, NIB_PAT, ERR_FLG) : novalue =
3458 ;          2695      begin
3459 ;          2696
3460 ;          2697  !++
3461 ;          2698  ! FUNCTIONAL DESCRIPTION:
3462 ;          2699  !       COMPARES THE CURRENT NIBBLE
3463 ;          2700  !       POSITION IN 'NIB_SAVE' WITH THE
3464 ;          2701  !       CURRENT TEST PATTERN.  IF THE
3465 ;          2702  !       TWO VALUES ARE NOT EQUAL AN
3466 ;          2703  !       ERROR FLG IS SET WHICH THE
3467 ;          2704  !       CALLER CAN INTERIGATE
3468 ;          2705  !
3469 ;          2706  ! FORMAL PARAMETERS:
3470 ;          2707  !       NIB_NUM
3471 ;          2708  !       CASE SELECT EXPRESSION TO SELECT THE
3472 ;          2709  !       CURRENT NIBBLE TO BE EXAMINED
3473 ;          2710  !
3474 ;          2711  !       NIB_PAT
3475 ;          2712  !       CURRENT NIBBLE PATTERN TO BE
3476 ;          2713  !       COMPARED
3477 ;          2714  !
3478 ;          2715  !       ERR_FLG
3479 ;          2716  !       CONTAINS THE ADDRESS (PASSED BY REF)
3480 ;          2717  !       OF THE CALLERS ERROR FLG
3481 ;          2718  !       TO ENABLE THE CALLER TO EXAMINE
3482 ;          2719  !       THE ERROR STATUS OF THE ROUTINE CALL
3483 ;          2720  !
3484 ;          2721  ! IMPLICIT INPUTS:
3485 ;          2722  !       NIB_SAVE
3486 ;          2723  !       BLOCK OF 3 WORDS TO STORE
3487 ;          2724  !       THE DATA FOUND IN MLD1, MLD2
3488 ;          2725  !       AND MLE2 AFTER A DIAGNOSTIC MODE
3489 ;          2726  !       READ
3490 ;          2727  !
3491 ;          2728  ! IMPLICIT OUTPUTS:     NONE
3492 ;          2729  !--
3493 ;          2730
3494 ;          2731      .ERR_FLG = ZERO;                          !CLEAR THE ERROR FLAG BACK IN THE CALLING ROUTINE
3495 ;          2732
3496 ;          2733      case .(NIB_NUM) from 0 to 9 of            !SELECT THE NIBBLE TO BE TESTED
3497 ;          2734          set
3498 ;          2735
3499 ;          2736          [0] :
3500 ;          2737
3501 ;          2738              if .NIB_SAVE [NIB_0] neq .(NIB_PAT)<0, 4> then .ERR_FLG = ONE;
3502 ;          2739
3503 ;          2740                                                !TEST NIBBLE 0 AND SET ERR FLG IF NEQ
3504 ;          2741
3505 ;          2742          [1] :
3506 ;          2743
3507 ;          2744              if .NIB_SAVE [NIB_1] neq .(NIB_PAT)<0, 4> then .ERR_FLG = ONE;
3508 ;          2745
```

```
3510 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
3511 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (16)
3512
3513 ;          2746                                         !TEST NIBBLE 1 AND SET ERR FLG IF NEQ
3514 ;          2747
3515 ;          2748        [2] :
3516 ;          2749
3517 ;          2750            if .NIB_SAVE [NIB_2] neq .(NIB_PAT)<0, 4> then .ERR_FLG = ONE;
3518 ;          2751
3519 ;          2752                                         !TEST NIBBLE 2 AND SET ERR FLG IF NEQ
3520 ;          2753
3521 ;          2754        [3] :
3522 ;          2755
3523 ;          2756            if .NIB_SAVE [NIB_3] neq .(NIB_PAT)<0, 4> then .ERR_FLG = ONE;
3524 ;          2757
3525 ;          2758                                         !TEST NIBBLE 3 AND SET ERR FLG IF NEQ
3526 ;          2759
3527 ;          2760        [4] .
3528 ;          2761
3529 ;          2762            if .NIB_SAVE [NIB_4] neq .(NIB_PAT)<0, 4> then .ERR_FLG = ONE;
3530 ;          2763
3531 ;          2764                                         !TEST NIBBLE 4 AND SET ERR FLG IF NEQ
3532 ;          2765
3533 ;          2766        [5] :
3534 ;          2767
3535 ;          2768            if .NIB_SAVE [NIB_5] neq .(NIB_PAT)<0, 4> then .ERR_FLG = ONE;
3536 ;          2769
3537 ;          2770                                         !TEST NIBBLE 5 AND SET ERR FLG IF NEQ
3538 ;          2771
3539 ;          2772        [6] :
3540 ;          2773
3541 ;          2774            if .NIB_SAVE [NIB_6] neq .(NIB_PAT)<0, 4> then .ERR_FLG = ONE;
3542 ;          2775
3543 ;          2776                                         !TEST NIBBLE 6 AND SET ERR FLG IF NEQ
3544 ;          2777
3545 ;          2778        [7] :
3546 ;          2779
3547 ;          2780            if .NIB_SAVE [NIB_7] neq .(NIB_PAT)<0, 4> then .ERR_FLG = ONE;
3548 ;          2781
3549 ;          2782                                         !TEST NIBBLE 7 AND SET ERR FLG IF NEQ
3550 ;          2783
3551 ;          2784        [8] :
3552 ;          2785
3553 ;          2786            if .NIB_SAVE [NIB_8] neq .(NIB_PAT)<0, 4> then .ERR_FLG = ONE;
3554 ;          2787
3555 ;          2788                                         !TEST NIBBLE 8 AND SET ERR FLG IF NEQ
3556 ;          2789
3557 ;          2790        [9] :
3558 ;          2791
3559 ;          2792            if .NIB_SAVE [NIB_9] neq .(NIB_PAT)<0, 3>
3560 ;          2793            then
3561 ;          2794                .ERR_FLG = ONE           !TEST NIBBLE 9 AND SET ERR FLG IF NEQ
3562 ;          2795        tes;
3563 ;          2796
3564 ;          2797    end;
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (16)

```
3566 ;ML4
3567 ;
3568
3572
3573                                        .SBTTL  TST.LNG.WRD
3577 016154                          TST.LNG.WRD:
3578 016154  004167  165664                JSR     R1,$SAVE2                   ;                                   2694
3579 016160  016600  000010                MOV     10(SP),R0                   ; ERR.FLG,*                         2731
3580 016164  005010                        CLR     (R0)
3581 016166  016601  000014                MOV     14(SP),R1                   ; NIB.NUM,*                         2733
3582 016172  006301                        ASL     R1
3583 016174  066107  016200                ADD     1$(R1),PC
3584 016200  000024                  1$:   .WORD   2$-1$
3585 016202  000052                        .WORD   3$-1$
3586 016204  000110                        .WORD   4$-1$
3587 016206  000140                        .WORD   5$-1$
3588 016210  000200                        .WORD   6$-1$
3589 016212  000226                        .WORD   7$-1$
3590 016214  000264                        .WORD   8$-1$
3591 016216  000314                        .WORD   9$-1$
3592 016220  000354                        .WORD   10$-1$
3593 016222  000404                        .WORD   11$-1$
3594 016224  016602  000012          2$:   MOV     12(SP),R2                   ; NIB.PAT,*                         2738
3595 016230  042702  177760                BIC     #177760,R2
3596 016234  016701  172712                MOV     NIB.SAVE,R1
3597 016240  042701  177760                BIC     #177760,R1
3598 016244  020102                        CMP     R1,R2
3599 016246  001577                        BEQ     13$
3600 016250  000574                        BR      12$
3601 016252  016601  000012          3$:   MOV     12(SP),R1                   ; NIB.PAT,*                         2744
3602 016256  042701  177760                BIC     #177760,R1
3603 016262  016702  172664                MOV     NIB.SAVE,R2
3604 016266  006202                        ASR     R2
3605 016270  006202                        ASR     R2
3606 016272  006202                        ASR     R2
3607 016274  006202                        ASR     R2
3608 016276  042702  177760                BIC     #177760,R2
3609 016302  020201                        CMP     R2,R1
3610 016304  001560                        BEQ     13$
3611 016306  000555                        BR      12$
3612 016310  016601  000012          4$:   MOV     12(SP),R1                   ; NIB.PAT,*                         2750
3613 016314  042701  177760                BIC     #177760,R1
3614 016320  016702  172626                MOV     NIB.SAVE,R2
3615 016324  000302                        SWAB    R2
3616 016326  042702  177760                BIC     #177760,R2
3617 016332  020201                        CMP     R2,R1
3618 016334  001544                        BEQ     13$
3619 016336  000541                        BR      12$
3620 016340  016601  000012          5$:   MOV     12(SP),R1                   ; NIB.PAT,*                         2756
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                                
016344  042701  177760                        BIC     #177760,R1
016350  016702  172576                        MOV     NIB.SAVE,R2
016354  006202                                ASR     R2
016356  006202                                ASR     R2
016360  006202                                ASR     R2
016362  006202                                ASR     R2
016364  000302                                SWAB    R2
016366  042702  177760                        BIC     #177760,R2
016372  020201                                CMP     R2,R1
016374  001524                                BEQ     13$
016376  000521                                BR      12$
016400  016601  000012                6$:     MOV     12(SP),R1               ; NIB.PAT,*                     2762
016404  042701  177760                        BIC     #177760,R1
016410  016702  172540                        MOV     NIB.SAVE+2,R2
016414  042702  177760                        BIC     #177760,R2
016420  020201                                CMP     R2,R1
016422  001511                                BEQ     13$
016424  000506                                BR      12$
016426  016601  000012                7$:     MOV     12(SP),R1               ; NIB.PAT,*                     2768
016432  042701  177760                        BIC     #177760,R1
016436  016702  172512                        MOV     NIB.SAVE+2,R2
016442  006202                                ASR     R2
016444  006202                                ASR     R2
016446  006202                                ASR     R2
016450  006202                                ASR     R2
016452  042702  177760                        BIC     #177760,R2
016456  020201                                CMP     R2,R1
016460  001472                                BEQ     13$
016462  000467                                BR      12$
016464  016601  000012                8$:     MOV     12(SP),R1               ; NIB.PAT,*                     2774
016470  042701  177760                        BIC     #177760,R1
016474  016702  172454                        MOV     NIB.SAVE+2,R2
016500  000302                                SWAB    R2
016502  042702  177760                        BIC     #177760,R2
016506  020201                                CMP     R2,R1
016510  001456                                BEQ     13$
016512  000453                                BR      12$
016514  016601  000012                9$:     MOV     12(SP),R1               ; NIB.PAT,*                     2780
016520  042701  177760                        BIC     #177760,R1
016524  016702  172424                        MOV     NIB.SAVE+2,R2
016530  006202                                ASR     R2
016532  006202                                ASR     R2
016534  006202                                ASR     R2
016536  006202                                ASR     R2
016540  000302                                SWAB    R2
016542  042702  177760                        BIC     #177760,R2
016546  020201                                CMP     R2,R1
016550  001436                                BEQ     13$
016552  000433                                BR      12$
016554  016601  000012                10$:    MOV     12(SP),R1               ; NIB.PAT,*                     2786
01656C  042701  177760                        BIC     #177760,R1
016564  016702  172366                        MOV     NIB.SAVE+4,R2
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<

016570  000302                                  SWAB    R2
016572  042702  177760                          BIC     #177760,R2
016576  020201                                  CMP     R2,R1
016600  001422                                  BEQ     13$
016602  000417                                  BR      12$
016604  016601  000012          11$:            MOV     12(SP),R1               ; NIB.PAT,*                     2792
016610  042701  177770                          BIC     #177770,R1
016614  016702  172336                          MOV     NIB.SAVE+4,R2
016620  006202                                  ASR     R2
016622  006202                                  ASR     R2
016624  006202                                  ASR     R2
016626  006202                                  ASR     R2
016630  000302                                  SWAB    R2
016632  042702  177770                          BIC     #177770,R2
016636  020201                                  CMP     R2,R1
016640  001402                                  BEQ     13$
016642  012710  000001          12$:            MOV     #1,(R0)                 ;                               2794
016646  000207                  13$:            RTS     PC                      ;                               2694

                                        ; Routine Size:  158 words
                                        ; Maximum stack depth per invocation:  3 words
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (17)

```
3709 ;ML4
3710 ;
3711
3712 ;          2798  routine XCR_LNG_WRD (NIB_NUM, NIB_PAT, RESULT) : novalue =
3713 ;          2799      begin
3714 ;          2800
3715 ;          2801  !++
3716 ;          2802  ! FUNCTIONAL DESCRIPTION:
3717 ;          2803  !       EXCLUSIVE ORS THE CURRENT
3718 ;          2804  !       NIBBLE POSITION IN 'NIB_SAVE'
3719 ;          2805  !       WITH THE CURRENT TEST PATTERN
3720 ;          2806  !       AND ASSIGNS THE RESULTS TO THE
3721 ;          2807  !       FORMAL PARAMETER 'RESULT'.
3722 ;          2808  !
3723 ;          2809  ! FORMAL PARAMETERS:
3724 ;          2810  !       NIB_NUM
3725 ;          2811  !       CASE SELECT EXPRESSION TO
3726 ;          2812  !       SELECT THE CURRENT NIBBLE TO BE
3727 ;          2813  !        EXAMINED
3728 ;          2814  !
3729 ;          2815  !       NIB_PAT
3730 ;          2816  !       CURRENT NIBBLE PATTERN TO BE
3731 ;          2817  !       XOR'ED
3732 ;          2818  !
3733 ;          2819  !       RESULT
3734 ;          2820  !       CONTAINS THE ADDRESS (PASSED BY REF)
3735 ;          2821  !       OF AN OWN STORAGE LOCATION TO
3736 ;          2822  !       ENABLE THE CALLER TO EXAMINE THE XOR RESULTS.
3737 ;          2823  !
3738 ;          2824  ! IMPLICIT INPUTS:
3739 ;          2825  !       NIB_SAVE
3740 ;          2826  !       BLOCK OF 3 WORDS TO STORE
3741 ;          2827  !       THE DATA FOUND IN MLD1
3742 ;          2828  !       MLD2 AND MLE2 AFTER A
3743 ;          2829  !       DIAGNOSTIC MODE READ.
3744 ;          2830  !
3745 ;          2831  ! IMPLICIT OUTPUTS:    NONE
3746 ;          2832  !--
3747 ;          2833
3748 ;          2834      case .(NIB_NUM) from 0 to 9 of              !SELECT THE NIBBLE TO BE XOR'ED
3749 ;          2835          set
3750 ;          2836
3751 ;          2837          [0] :
3752 ;          2838              .RESULT = .NIB_SAVE [NIB_0] xor .NIB_PAT;
3753 ;          2839                                          !XOR NIBBLE 0 AND STORE RESULTS IN THE ADRS CONTAINED IN .RESULTS
3754 ;          2840
3755 ;          2841          [1] :
3756 ;          2842              .RESULT = .NIB_SAVE [NIB_1] xor .NIB_PAT;
3757 ;          2843                                          !XOR NIBBLE 1 AND STORE RESULTS IN THE ADRS CONTAINED IN .RESULTS
3758 ;          2844
3759 ;          2845          [2] :
3760 ;          2846              .RESULT = .NIB_SAVE [NIB_2] xor .NIB_PAT;
3761 ;          2847                                          !XOR NIBBLE 2 AND STORE RESULTS IN THE ADRS CONTAINED IN .RESULTS
3762 ;          2848
3763 ;          2849          [3] :
```

```
                                                        14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
                                                        14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (17)

;ML4
;
;          2850              .RESULT = .NIB_SAVE [NIB_3] xor .NIB_PAT;
;          2851                                          !XOR NIBBLE 3 AND STORE RESULTS IN THE ADRS CONTAINED IN .RESULTS
;          2852
;          2853          [4] :
;          2854              .RESULT = .NIB_SAVE [NIB_4] xor .NIB_PAT;
;          2855                                          !XOR NIBBLE 4 AND STORE RESULTS IN THE ADRS CONTAINED IN .RESULTS
;          2856
;          2857          [5] :
;          2858              .RESULT = .NIB_SAVE [NIB_5] xor .NIB_PAT;
;          2859                                          !XOR NIBBLE 5 AND STORE RESULTS IN THE ADRS CONTAINED IN .RESULTS
;          2860
;          2861          [6] :
;          2862              .RESULT = .NIB_SAVE [NIB_6] xor .NIB_PAT;
;          2863                                          !XOR NIBBLE 6 AND STORE RESULTS IN THE ADRS CONTAINED IN .RESULTS
;          2864
;          2865          [7] :
;          2866              .RESULT = .NIB_SAVE [NIB_7] xor .NIB_PAT;
;          2867                                          !XOR NIBBLE 7 AND STORE RESULTS IN THE ADRS CONTAINED IN .RESULTS
;          2868
;          2869          [8] :
;          2870              .RESULT = .NIB_SAVE [NIB_8] xor .NIB_PAT;
;          2871                                          !XOR NIBBLE 8 AND STORE RESULTS IN THE ADRS CONTAINED IN .RESULTS
;          2872
;          2873          [9] :
;          2874              .RESULT = .NIB_SAVE [NIB_9] xor .NIB_PAT
;          2875                                          !XOR NIBBLE 9 AND STORE RESULTS IN THE ADRS CONTAINED IN .RESULTS
;          2876          tes;
;          2877
;          2878      end;
3800
3801                                  .SBTTL  XOR.LNG.WRD
3805 016650                     XOR.LNG.WRD:
3806 016650  004167  165222              JSR     R1,$SAVE4                ;                                        2798
3807 016654  016601  000016              MOV     16(SP),R1                ; NIB.PAT,*                              2838
3808 016660  016602  000014              MOV     14(SP),R2                ; RESULT,*
3809 016664  016600  000020              MOV     20(SP),R0                ; NIB.NUM,*                              2834
3810 016670  006300                      ASL     R0
3811 016672  066007  016676              ADD     1$(R0),PC
3812 016676  000024              1$:     .WORD   2$-1$
3813 016700  000036                      .WORD   3$-1$
3814 015702  000062                      .WORD   4$-1$
3815 016704  000070                      .WORD   5$-1$
3816 016706  000076                      .WORD   6$-1$
3817 016710  000120                      .WORD   8$-1$
3818 016712  000136                      .WORD   9$-1$
3819 016714  000144                      .WORD   10$-1$
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                                .WORD   12$-1$
                                                .WORD   15$-1$
016722  016712  172224        2$:     MOV     NIB.SAVE,(R2)           ;                       2838
016726  042712  177760                BIC     #177760,(R2)
016732  000472                        BR      17$
016734  016703  172212        3$:     MOV     NIB.SAVE,R3             ;                       2842
016740  006203                        ASR     R3
016742  006203                        ASR     R3
016744  006203                        ASR     R3
016746  006203                        ASR     R3
016750  042703  177760                BIC     #177760,R3
016754  010312                        MOV     R3,(R2)
016756  000412                        BR      7$
016760  016704  172166        4$:     MOV     NIB.SAVE,R4             ;                       2846
016764  000437                        BR      13$
016766  016704  172160        5$:     MOV     NIB.SAVE,R4             ;                       2850
016772  000425                        BR      11$
016774  016712  172154        6$:     MOV     NIB.SAVE+2,(R2)         ;                       2854
017000  042712  177760                BIC     #177760,(R2)
017004  010104                7$:     MOV     R1,R4
017006  041204                        BIC     (R2),R4
017010  040112                        BIC     R1,(R2)
017012  050412                        BIS     R4,(R2)
017014  000207                        RTS     PC                      ;                       2834
017016  016704  172132        8$:     MOV     NIB.SAVE+2,R4           ;                       2858
017022  006204                        ASR     R4
017024  006204                        ASR     R4
017026  006204                        ASR     R4
017030  006204                        ASR     R4
017032  000415                        BR      14$
017034  016704  172114        9$:     MOV     NIB.SAVE+2,R4           ;                       2862
017040  000411                        BR      13$
017042  016704  172106        10$:    MOV     NIB.SAVE+2,R4           ;                       2866
017046  006204                11$:    ASR     R4
017050  006204                        ASR     R4
017052  006204                        ASR     R4
017054  006204                        ASR     R4
017056  000402                        BR      13$
017060  016704  172072        12$:    MOV     NIB.SAVE+4,R4           ;                       2870
017064  000304                13$:    SWAB    R4
017066  042704  177760        14$:    BIC     #177760,R4
017072  000411                        BR      16$
017074  016704  172056        15$:    MOV     NIB.SAVE+4,R4           ;                       2874
017100  006204                        ASR     R4
017102  006204                        ASR     R4
017104  006204                        ASR     R4
017106  006204                        ASR     R4
017110  000304                        SWAB    R4
017112  042704  177770                BIC     #177770,R4
017116  010412                16$:    MOV     R4,(R2)
017120  010103                17$:    MOV     R1,R3
017122  041203                        BIC     (R2),R3
```

```
                                ;ML4                                                    14-May-1981 13:48:54    TOPS
                                ;                                                       14-May-1981 13:05:08    PA:<

017124  040112                          BIC     R1,(R2)
017126  050312                          BIS     R3,(R2)
017130  000207                          RTS     PC                              ;                               2798

                                ; Routine Size:  89 words
                                ; Maximum stack depth per invocation:  5 words
```

```
3893 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
3894 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (18)
3895
3896 ;          2879  routine LD_LNG_WRD (NIB_NUM, NIB_PAT) : novalue =
3897 ;          2880      begin
3898 ;          2881
3899 ;          2882  !++
3900 ;          2883  ! FUNCTIONAL DESCRIPTION:
3901 ;          2884  !       LOADS 'NIB_SAVE' WITH UNIQUE
3902 ;          2885  !       NIBBLE PATTERNS PRIOR TO WRITING
3903 ;          2886  !       TO MLD1, MLD2 AND MLE2
3904 ;          2887  !       DATA DIAGNOSTIC REGISTERS.
3905 ;          2888  !
3906 ;          2889  ! FORMAL PARAMETERS:
3907 ;          2890  !       NIB_NUM
3908 ;          2891  !       CASE SELECT EXPRESSION TO SELECT
3909 ;          2892  !       THE CURRENT NIBBLE TO BE LOADED
3910 ;          2893  !
3911 ;          2894  !       NIB_PAT
3912 ;          2895  !       CURRENT NIBBLE PATTERN TO BE
3913 ;          2896  !       LOADED
3914 ;          2897  !
3915 ;          2898  ! IMPLICIT INPUTS:
3916 ;          2899  !       NIB_SAVE
3917 ;          2900  !       BLOCK OF 3 WORDS TO STORE
3918 ;          2901  !       THE DATA TO BE WRITTEN
3919 ;          2902  !       INTO MLD1 MLD2 MLE2
3920 ;          2903  !
3921 ;          2904  ! IMPLICIT OUTPUTS:
3922 ;          2905  !       NIB_SAVE IS LOADED WIH
3923 ;          2906  !       THE CURRENT NIBBLE PATTERN
3924 ;          2907  !--
3925 ;          2908
3926 ;          2909      case .(NIB_NUM) from 0 to 9 of          !SELECT THE NIBBLE LOCATION IN NIB_SAVE TO BE LOADED
3927 ;          2910      set
3928 ;          2911
3929 ;          2912          [0] :
3930 ;          2913              NIB_SAVE [NIB_0] = .NIB_PAT;    !LOAD NIBBLE 0 WITH NIB_PAT
3931 ;          2914
3932 ;          2915          [1] :
3933 ;          2916              NIB_SAVE [NIB_1] = .NIB_PAT;    !LOAD NIBBLE 1 WITH NIB_PAT
3934 ;          2917
3935 ;          2918          [2] :
3936 ;          2919              NIB_SAVE [NIB_2] = .NIB_PAT;    !LOAD NIBBLE 2 WITH NIB_PAT
3937 ;          2920
3938 ;          2921          [3] :
3939 ;          2922              NIB_SAVE [NIB_3] = .NIB_PAT;    'LOAD NIBBLE 3 WITH NIB_PAT
3940 ;          2923
3941 ;          2924          [4] :
3942 ;          2925              NIB_SAVE [NIB_4] = .NIB_PAT;    :LOAD NIBBLE 4 WITH NIB_PAT
3943 ;          2926
3944 ;          2927          [5] :
3945 ;          2928              NIB_SAVE [NIB_5] = .NIB_PAT;    !LOAD NIBBLE 5 WITH NIB_PAT
3946 ;          2929
3947 ;          2930          [6] :
```

```
                                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
                                                                    14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (18)
;ML4
;
;          2931              NIB_SAVE [NIB_6] = .NIB_PAT;          !LOAD NIBBLE 6 WITH NIB_PAT
;          2932
;          2933          [7] :
;          2934              NIB_SAVE [NIB_7] = .NIB_PAT;          !LOAD NIBBLE 7 WITH NIB_PAT
;          2935
;          2936          [8] :
;          2937              NIB_SAVE [NIB_8] = .NIB_PAT;          !LOAD NIBBLE 8 WITH NIB_PAT
;          2938
;          2939          [9] :
;          2940              NIB_SAVE [NIB_9] = .NIB_PAT           !LOAD NIBBLE 9 WITH NIB_PAT
;          2941          tes;
;          2942
;          2943      end;

                                    .SBTTL  LD.LNG.WRD
017132                        LD.LNG.WRD:
017132  010146                       MOV     R1,-(SP)                ;                                       2879
017134  016600  000004               MOV     4(SP),R0                ; NIB.PAT,*                             2913
017140  016601  000006               MOV     6(SP),R1                ; NIB.NUM,*                             2909
017144  006301                       ASL     R1
017146  066107  017152               ADD     1$(R1),PC
017152  000024                1$:    .WORD   2$-1$
017154  000040                       .WORD   3$-1$
017156  000064                       .WORD   4$-1$
017160  000102                       .WORD   5$-1$
017162  000134                       .WORD   7$-1$
017164  000150                       .WORD   8$-1$
017166  000174                       .WORD   9$-1$
017170  000212                       .WORD   10$-1$
017172  000244                       .WORD   12$-1$
017174  000262                       .WORD   13$-1$
017176  042700  177760        2$:    BIC     #177760,R0              ;                                       2913
017202  142767  000017  171742       BICB    #17,NIB.SAVE
017210  000433                       BR      6$
017212  006300                3$:    ASL     R0                      ;                                       2916
017214  006300                       ASL     R0
017216  006300                       ASL     R0
017220  006300                       ASL     R0
017222  042700  177417               BIC     #177417,R0
017226  142767  000360  171716       BICB    #360,NIB.SAVE
017234  000421                       BR      6$
017236  000300                4$:    SWAB    R0                      ;                                       2919
017240  042700  170377               BIC     #170377,R0
017244  042767  007400  171700       BIC     #7400,NIB.SAVE
017252  000412                       BR      6$
017254  000300                5$:    SWAB    R0                      ;                                       2922
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                                ASL     R0
                                                ASL     R0
                                                ASL     R0
                                                ASL     R0
                                                BIC     #7777,R0
                                                BIC     #170000,NIB.SAVE
                                        6$:     BIS     R0,NIB.SAVE
                                                BR      15$                             ;                               2909
                                        7$:     BIC     #177760,R0                      ;                               2925
                                                BICB    #17,NIB.SAVE+2
                                                BR      11$
                                        8$:     ASL     R0                              ;                               2928
                                                ASL     R0
                                                ASL     R0
                                                ASL     R0
                                                BIC     #177417,R0
                                                BICB    #360,NIB.SAVE+2
                                                BR      11$
                                        9$:     SWAB    R0                              ;                               2931
                                                BIC     #170377,R0
                                                BIC     #7400,NIB.SAVE+2
                                                BR      11$
                                        10$:    SWAB    R0                              ;                               2934
                                                ASL     R0
                                                ASL     R0
                                                ASL     R0
                                                ASL     R0
                                                BIC     #7777,R0
                                                BIC     #170000,NIB.SAVE+2
                                        11$:    BIS     R0,NIB.SAVE+2
                                                BR      15$                             ;                               2909
                                        12$:    SWAB    R0                              ;                               2937
                                                BIC     #170377,R0
                                                BIC     #7400,NIB.SAVE+4
                                                BR      14$
                                        13$:    SWAB    R0                              ;                               2940
                                                ASL     R0
                                                ASL     R0
                                                ASL     R0
                                                ASL     R0
                                                BIC     #107777,R0
                                                BIC     #70000,NIB.SAVE+4
                                        14$:    BIS     R0,NIB.SAVE+4
                                        15$:    MOV     (SP)+,R1                        ;                               2879
                                                RTS     PC

                                        ; Routine Size:  111 words
                                        ; Maximum stack depth per invocation:  2 words
```

```
4061 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
4062 :                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (19)
4063
4064 :          2944  routine WRT_CS1 (TST_PAT, index) : novalue =
4065 :          2945      begin
4066 :          2946
4067 :          2947  !++
4068 :          2948  ! FUNCTIONAL DESCRIPTION:
4069 :          2949  !
4070 :          2950  !       LOADS THE CONTROL & STATUS REGISTER 1 WITH A DATA PATTERN
4071 :          2951  !       GENERATED BY THE MACRO WRT_MASK.
4072 :          2952  !
4073 :          2953  ! FORMAL PARAMETERS:
4074 :          2954  !       TST_PAT
4075 :          2955  !       CURRENT DATA PATTERN TO BE LOADED IN THE REGISTER.
4076 :          2956  !
4077 :          2957  !       INDEX
4078 :          2958  !       USED BY THE MACRO WRT_MASK TO SELECT THE CURRENT REGISTERS ADDRESS,
4079 :          2959  !       FORCED HI, FORCED LO AND DON'T CARE MASK INFORMATION.
4080 :          2960  !
4081 :          2961  !--
4082 :          2962
4083 :          2963      MLCS1 = WRT_MASK;                          !LOAD GENERATED WRT_MASK PATTERN INTO MLCS1
4084 :          2964      end;
4088
4089                                    .SBTTL  WRT.CS1
4093 017470  004167  164350       WRT.CS1:JSR     R1,$SAVE2                       ;                          2944
4094 017474  016600  000010               MOV     10(SP),R0                       ; INDEX,*                  2963
4095 017500  006300                       ASL     R0
4096 017502  006300                       ASL     R0
4097 017504  006300                       ASL     R0
4098 017506  010001                       MOV     R0,R1
4099 017510  016100  013156               MOV     ML.REG+2(R1),R0
4100 017514  056600  000012               BIS     12(SP),R0                       ; TST.PAT,*
4101 017520  046100  013160               BIC     ML.REG+4(R1),R0
4102 017524  016102  013162               MOV     ML.REG+6(R1),R2
4103 017530  050002                       BIS     R0,R2
4104 017532  010277  173416               MOV     R2,@ML.REG
4105 017536  000207                       RTS     PC                              ;                          2944
4106
4107                                ; Routine Size:  20 words
4108                                ; Maximum stack depth per invocation:  3 words
4113
4114
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (20)

```
4116 ;ML4
4117 ;
4118
4119 ;          2965  routine RD_CS1 (TST_PAT, index, ERR_FLG) : novalue =
4120 ;          2966      begin
4121 ;          2967
4122 ;          2968  !++
4123 ;          2969  ! FUNCTIONAL DESCRIPTION:
4124 ;          2970  !
4125 ;          2971  !       COMPARES THE CONTENTS OF THE
4126 ;          2972  !       CONTROL & STATUS REGISTER 1
4127 ;          2973  !       WITH THE MASKED DATA PATTERN
4128 ;          2974  !       GENERATED BY THE MACRO 'WRT_MASK'
4129 ;          2975  !
4130 ;          2976  !       IF THE COMPARE IS NOT EQUAL THEN
4131 ;          2977  !       THE FORMAL PARAMETER 'ERR_FLG' IS
4132 ;          2978  !       ASSIGNED A ONE TO INDICATE THE
4133 ;          2979  !       ERROR.
4134 ;          2980  !
4135 ;          2981  ! FORMAL PARAMETERS:
4136 ;          2982  !
4137 ;          2983  !       TST_PAT
4138 ;          2984  !       DATA PATTERN TO BE MASKED AND
4139 ;          2985  !       COMPARED AGAINST THE CONTENTS
4140 ;          2986  !       OF THE REGISTER UNDER TEST.
4141 ;          2987  !
4142 ;          2988  !       INDEX
4143 ;          2989  !        USED BY THE MACRO WRT_MASK TO
4144 ;          2990  !       SELECT THE CURRENT REGISTERS ADDRESS,
4145 ;          2991  !       FORCED HI, FORCED LO AND DON'T CARE
4146 ;          2992  !       MASK INFORMATION.
4147 ;          2993  !
4148 ;          2994  !       ERR_FLG
4149 ;          2995  !       CONTAINS THE ADDRESS (PASSED BY REF)
4150 ;          2996  !       OF THE CALLERS ERROR FLG TO ENABLE THE
4151 ;          2997  !       CALLER TO EXAMINE THE ERROR STATUS
4152 ;          2998  !       OF THE ROUTINE CALL.
4153 ;          2999  !
4154 ;          3000  ! IMPLICIT INPUTS:
4155 ;          3001  !       WT_DATA
4156 ;          3002  !         GETS LOADED WITH THE GENERATED
4157 ;          3003  !         WRT_MASK DATA PATTERN THUS ALLOWING
4158 ;          3004  !         CALLER TO PRINT FAILING GOOD DATA.
4159 ;          3005  !
4160 ;          3006  !       RD_DATA
4161 ;          3007  !         GETS LOADED WITH DATA READ FROM THE
4162 ;          3008  !         REGISTER THUS ALLOWING CALLER
4163 ;          3009  !         TO PRINT FAILING BAD DATA.
4164 ;          3010  !
4165 ;          3011  ! IMPLICIT OUTPUTS:
4166 ;          3012  !       GLOBAL LOCATION WR_DATA
4167 ;          3013  !       AND RD_DATA LOADED WITH GOOD
4168 ;          3014  !       AND BAD REGISTER DATA
4169 ;          3015  !--
4170 ;          3016
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 v2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (20)

;          3017      .ERR_FLG = ZERO;                   !CLEAR ERROR FLAG
;          3018      WT_DATA = WRT_MASK;                !SAVE THE DATA WRITTEN TO THE REGISTER
;          3019      RD_DATA = .ML.CS1 or .IGNORE;      !READ AND SAVE THE REGISTER
;          3020
;          3021      if .WT_DATA neq .RD_DATA then .ERR_FLG = ONE;       !READ THE REG FOR WRT_MASK
;          3022
;          3023                                         !SET ERROR FLAG IF NEQ
;          3024      end;

                                    .SBTTL  RD.CS1
017540 004167 164300          RD.CS1: JSR     R1,$SAVE2                       ;                                2965
017544 005076 000010                  CLR     @10(SP)                         ; ERR.FLG                        3017
017550 016600 000012                  MOV     12(SP),R0                       ; INDEX,*                        3018
017554 006300                         ASL     R0
017556 006300                         ASL     R0
017560 006300                         ASL     R0
017562 010001                         MOV     R0,R1
017564 016100 013156                  MOV     ML.REG+2(R1),R0
017570 056600 000014                  BIS     14(SP),R0                       ; TST.PAT,*
017574 046100 013160                  BIC     ML.REG+4(R1),R0
017600 016102 013162                  MOV     ML.REG+6(R1),R2
017604 050002                         BIS     R0,R2
017606 010267 173314                  MOV     R2,WT.DATA
017612 017702 173336                  MOV     @ML.REG,R2                      ;                                3019
017616 056102 013162                  BIS     ML.REG+6(R1),R2
017622 010267 173302                  MOV     R2,RD.DATA
017626 026767 173274 173274           CMP     WT.DATA,RD.DATA                 ;                                3021
017634 001403                         BEQ     1$
017636 012776 000001 000010           MOV     #1,@10(SP)                      ; *,ERR.FLG
017644 000207                 1$:     RTS     PC                              ;                                2965

                                    ; Routine Size:  35 words
                                    ; Maximum stack depth per invocation:  3 words
```

```
4221 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
4222 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (21)
4223
4224 ;          3025  routine WRT_ER (TST_PAT, index) : novalue =
4225 ;          3026      begin
4226 ;          3027
4227 ;          3028  !++
4228 ;          3029  ! FUNCTIONAL DESCRIPTION:
4229 ;          3030  !
4230 ;          3031  !       LOADS THE ERROR REGISTER WITH A DATA PATTERN GENERATED BY THE MACRO WRT_MASK
4231 ;          3032  !
4232 ;          3033  ! FORMAL PARAMETERS:
4233 ;          3034  !
4234 ;          3035  !       TST_PAT
4235 ;          3036  !       CURRENT DATA PATTERN TO BE LOADED IN THE REGISTER.
4236 ;          3037  !
4237 ;          3038  !       INDEX
4238 ;          3039  !       USED BY THE MACRO WRT_MASK TO SELECT THE CURRENT REGISTERS ADDRESS,
4239 ;          3040  !       FORCED HI, FORCED LO AND DON'T CARE MASK INFORMATION.
4240 ;          3041  !--
4241 ;          3042
4242 ;          3043      MLER = WRT_MASK;                          !LOAD GENERATE WRT_MASK PATTERN INTO MLER
4243 ;          3044      end;
4247
4248                                        .SBTTL  WRT.ER
4252 017646  004167  164172          WRT.ER: JSR     R1,$SAVE2                       ;                       3025
4253 017652  016600  000010                  MOV     10(SP),R0                       ; INDEX,*               3043
4254 017656  006300                          ASL     R0
4255 017660  006300                          ASL     R0
4256 017662  006300                          ASL     R0
4257 017664  010001                          MOV     R0,R1
4258 017666  016100  013156                  MOV     ML.REG+2(R1),R0
4259 017672  056600  000012                  BIS     12(SP),R0                       ; TST.PAT,*
4260 017676  046100  013160                  BIC     ML.REG+4(R1),R0
4261 017702  016102  013162                  MOV     ML.REG+6(R1),R2
4262 017706  050002                          BIS     R0,R2
4263 017710  010277  173320                  MOV     R2,@ML.REG+60
4264 017714  000207                          RTS     PC                              ;                       3025
4265
4266                                ; Routine Size:  20 words
4267                                ; Maximum stack depth per invocation:  3 words
4272
4273
```

```
4275 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
4276 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (22)
4277
4278 ;          3045  routine RD_ER (TST_PAT, index, ERR_FLG) : novalue =
4279 ;          3046      begin
4280 ;          3047
4281 ;          3048  !++
4282 ;          3049  ! FUNCTIONAL DESCRIPTION:
4283 ;          3050  !
4284 ;          3051  !       COMPARES THE CONTENTS OF THE
4285 ;          3052  !       ERROR REGISTER WITH
4286 ;          3053  !       THE MASKED DATA PATTERN
4287 ;          3054  !       GENERATED BY THE MACRO 'WRT_MASK'.
4288 ;          3055  !
4289 ;          3056  !       IF THE COMPARE IS NOT EQUAL THEN
4290 ;          3057  !       THE FORMAL PARAMETER 'ERR_FLG' IS
4291 ;          3058  !       ASSIGNED A ONE TO INDICATE THE
4292 ;          3059  !       ERROR.
4293 ;          3060  !
4294 ;          3061  ! FORMAL PARAMETERS:
4295 ;          3062  !
4296 ;          3063  !       TST_PAT
4297 ;          3064  !       DATA PATTERN TO BE MASKED AND
4298 ;          3065  !       COMPARED AGAINST THE CONTENTS
4299 ;          3066  !       OF THE REGISTER UNDER TEST.
4300 ;          3067  !
4301 ;          3068  !       INDEX
4302 ;          3069  !       USED BY THE MACRO WRT_MASK TO
4303 ;          3070  !       SELECT THE CURRENT REGISTERS ADDRESS,
4304 ;          3071  !       FORCED HI, FORCED LO AND DON'T CARE
4305 ;          3072  !       MASK INFORMATION.
4306 ;          3073  !
4307 ;          3074  !       ERR_FLG
4308 ;          3075  !       CONTAINS THE ADDRESS (PASSED BY REF)
4309 ;          3076  !       OF THE CALLERS ERROR FLG TO ENABLE THE
4310 ;          3077  !       CALLER TO EXAMINE THE ERROR STATUS
4311 ;          3078  !       OF THE ROUTINE CALL.
4312 ;          3079  !
4313 ;          3080  ! IMPLICIT INPUTS:
4314 ;          3081  !       WT_DATA
4315 ;          3082  !         GETS LOADED WITH THE GENERATED
4316 ;          3083  !         WRT_MASK DATA PATTERN THUS ALLOWING
4317 ;          3084  !         CALLER TO PRINT FAILING GOOD DATA.
4318 ;          3085  !
4319 ;          3086  !       RD_DATA
4320 ;          3087  !         GETS LOADED WITH DATA READ FROM THE
4321 ;          3088  !         REGISTER THUS ALLOWING CALLER
4322 ;          3089  !         TO PRINT FAILING BAD DATA.
4323 ;          3090  !
4324 ;          3091  ! IMPLICIT OUTPUTS:
4325 ;          3092  !       GLOBAL LOCATION WR_DATA
4326 ;          3093  !       AND RD_DATA LOADED WITH GOOD
4327 ;          3094  !       AND BAD REGISTER DATA
4328 ;          3095  !
4329 ;          3096  !--
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (22)

;          3097
;          3098        .ERR_FLG = ZERO;                          !CLEAR THE ERROR FLAG
;          3099        WT_DATA = WRT_MASK;                       !SAVE THE DATA WRITTEN TO THE REGISTER
;          3100        RD_DATA = .MLER or .IGNORE;               !READ AND SAVE THE REGISTER
;          3101
;          3102        if .WT_DATA neq .RD_DATA then .ERR_FLG = ONE;       !READ THE REG FOR WRT_MASK
;          3103
;          3104                                                  !SET ERROR FLAG IF NEQ
;          3105        end;

                                        .SBTTL  RD.ER
017716  004167  164122          RD.ER:  JSR     R1,$SAVE2                       ;                       3045
017722  005076  000010                  CLR     @10(SP)                         ; ERR.FLG               3098
017726  016600  000012                  MOV     12(SP),R0                       ; INDEX,*               3099
017732  006300                          ASL     R0
017734  006300                          ASL     R0
017736  006300                          ASL     R0
017740  010001                          MOV     R0,R1
017742  016100  013156                  MOV     ML.REG+2(R1),R0
017746  056600  000014                  BIS     14(SP),R0                       ; TST.PAT,*
017752  046100  013160                  BIC     ML.REG+4(R1),R0
017756  016102  013162                  MOV     ML.REG+6(R1),R2
017762  050002                          BIS     R0,R2
017764  010267  173136                  MOV     R2,WT.DATA
017770  017702  173240                  MOV     @ML.REG+60,R2                   ;                       3100
017774  056102  013162                  BIS     ML.REG+6(R1),R2
020000  010267  173124                  MOV     R2,RD.DATA
020004  026767  173116  173116          CMP     WT.DATA,RD.DATA                 ;                       3102
020012  001403                          BEQ     1$
020014  012776  000001  000010          MOV     #1,@10(SP)                      ; *,ERR.FLG
020022  000207                  1$:     RTS     PC                              ;                       3045

                                ; Routine Size:  35 words
                                ; Maximum stack depth per invocation:  3 words
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
:                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (23)

                3106  routine WRT_DA (TST_PAT, index) : novalue =
                3107      begin
                3108
                3109  !++
                3110  ! FUNCTIONAL DESCRIPTION:
                3111  !
                3112  !       LOADS THE DESIRED SECTOR WITH A DATA PATTERN GENERATED BY THE MACRO WRT_MASK
                3113  !
                3114  ! FORMAL PARAMETERS:
                3115  !
                3116  !       TST_PAT
                3117  !       CURRENT DATA PATTERN TO BE LOADED IN THE REGISTER.
                3118  !
                3119  !       INDEX
                3120  !       USED BY THE MACRO WRT_MASK TO SELECT THE CURRENT REGISTERS ADDRESS,
                3121  !       FORCED HI, FORCED LO AND DON'T CARE MASK INFORMATION.
                3122  !--
                3123
                3124      MLDA = WRT_MASK;                          !LOAD MLDA WITH GENERATED WRT_MASK PATTERN
                3125      end;

                                        .SBTTL  WRT.DA
020024  004167  164014          WRT.DA: JSR     R1,$SAVE2                       ;                               3106
020030  016600  000010                  MOV     10(SP),R0                       ; INDEX,*                       3124
020034  006300                          ASL     R0
020036  006300                          ASL     R0
020040  006300                          ASL     R0
020042  010001                          MOV     R0,R1
020044  016100  013156                  MOV     ML.REG+2(R1),R0
020050  056600  000012                  BIS     12(SP),R0                       ; TST.PAT,*
020054  046100  013160                  BIC     ML.REG+4(R1),R0
020060  016102  013162                  MOV     ML.REG+6(R1),R2
020064  050002                          BIS     R0,R2
020066  010277  173112                  MOV     R2,@ML.REG+30
020072  000207                          RTS     PC                              ;                               3106

                                ; Routine Size:  20 words
                                ; Maximum stack depth per invocation:  3 words
```

```
4435 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
4436 :                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (24)
4437
4438 ;          3126  routine RD_DA (TST_PAT, index, ERR_FLG) : novalue =
4439 ;          3127      begin
4440 ;          3128
4441 ;          3129  !++
4442 ;          3130  ! FUNCTIONAL DESCRIPTION:
4443 ;          3131  !
4444 ;          3132  !       COMPARES THE CONTENTS OF THE
4445 ;          3133  !       DESIRED SECTOR ADDRESS REGISTER
4446 ;          3134  !       WITH THE MASKED DATA PATTERN
4447 ;          3135  !       GENERATED BY THE MACRO 'WRT_MASK'
4448 ;          3136  !
4449 ;          3137  !       IF THE COMPARE IS NOT EQUAL THEN
4450 ;          3138  !       THE FORMAL PARAMETER 'ERR_FLG' IS
4451 ;          3139  !       ASSIGNED A ONE TO INDICATE THE
4452 ;          3140  !       ERROR
4453 ;          3141  !
4454 ;          3142  ! FORMAL PARAMETERS:
4455 ;          3143  !
4456 ;          3144  !       TST_PAT
4457 ;          3145  !       DATA PATTERN TO BE MASKED AND
4458 ;          3146  !       COMPARED AGAINST THE CONTENTS
4459 ;          3147  !       OF THE REGISTER UNDER TEST
4460 ;          3148  !
4461 ;          3149  !       INDEX
4462 ;          3150  !       USED BY THE MACRO WRT_MASK TO
4463 ;          3151  !       SELECT THE CURRENT REGISTERS ADDRESS,
4464 ;          3152  !       FORCED HI, FORCED LO AND DON'T CARE
4465 ;          3153  !       MASK INFORMATION
4466 ;          3154  !
4467 ;          3155  !       ERR_FLG
4468 ;          3156  !       CONTAINS THE ADDRESS (PASSED BY REF)
4469 ;          3157  !       OF THE CALLERS ERROR FLG TO ENABLE THE
4470 ;          3158  !       CALLER TO EXAMINE THE ERROR STATUS
4471 ;          3159  !       OF THE ROUTINE CALL.
4472 ;          3160  ! IMPLICIT INPUTS:
4473 ;          3161  !       WT_DATA
4474 ;          3162  !         GETS LOADED WITH THE GENERATED
4475 ;          3163  !         WRT_MASK DATA PATTERN THUS ALLOWING
4476 ;          3164  !         CALLER TO PRINT FAILING GOOD DATA.
4477 ;          3165  !
4478 ;          3166  !       RD_DATA
4479 ;          3167  !         GETS LOADED WITH DATA READ FROM THE
4480 ;          3168  !         REGISTER THUS ALLOWING CALLER
4481 ;          3169  !         TO PRINT FAILING BAD DATA.
4482 ;          3170  !
4483 ;          3171  ! IMPLICIT OUTPUTS:
4484 ;          3172  !       GLOBAL LOCATION WR_DATA
4485 ;          3173  !       AND RD_DATA LOADED WITH GOOD
4486 ;          3174  !       AND BAD REGISTER DATA
4487 ;          3175  !
4488 ;          3176  !--
4489 ;          3177
```

```
 4491 ;ML4                                                          14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
 4492 ;                                                             14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (24)
 4493
 4494 ;          3178      .ERR_FLG = ZERO;                         !CLEAR THE ERROR FLAG
 4495 ;          3179      WT_DATA = WRT_MASK;                      !SAVE THE DATA WRITTEN TO THE REGISTER
 4496 ;          3180      RD_DATA = .ML_DA or .IGNORE;             !READ AND SAVE THE REGISTER
 4497 ;          3181
 4498 ;          3182      if .WT_DATA neq .RD_DATA then .ERR_FLG = ONE;       !READ THE REG FOR WRT_MASK
 4499 ;          3183
 4500 ;          3184                                               !SET ERROR FLAG IF NEQ
 4501 ;          3185      end;
 4505
 4506                                        .SBTTL  RD.DA
 4510 020074  004167  163744          RD.DA: JSR     R1,$SAVE2                ;                          3126
 4511 020100  005076  000010                 CLR     @10(SP)                  ; ERR.FLG                  3178
 4512 020104  016600  000012                 MOV     12(SP),R0                ; INDEX,*                  3179
 4513 020110  006300                         ASL     R0
 4514 020112  006300                         ASL     R0
 4515 020114  006300                         ASL     R0
 4516 020116  010001                         MOV     R0,R1
 4517 020120  016100  013156                 MOV     ML.REG+2(R1),R0
 4518 020124  056600  000014                 BIS     14(SP),R0                ; TST.PAT,*
 4519 020130  046100  013160                 BIC     ML.REG+4(R1),R0
 4520 020134  016102  013162                 MOV     ML.REG+6(R1),R2
 4521 020140  050002                         BIS     R0,R2
 4522 020142  010267  172760                 MOV     R2,WT.DATA
 4523 020146  017702  173032                 MOV     @ML.REG+30,R2            ;                          3180
 4524 020152  056102  013162                 BIS     ML.REG+6(R1),R2
 4525 020156  010267  172746                 MOV     R2,RD.DATA
 4526 020162  026767  172740  172740        CMP     WT.DATA,RD.DATA          ;                          3182
 4527 020170  001403                         BEQ     1$
 4528 020172  012776  000001  000010        MOV     #1,@10(SP)               ; *,ERR.FLG
 4529 020200  000207                 1$:     RTS     PC                       ;                          3126
 4530
 4531                                        ; Routine Size:  35 words
 4532                                        ; Maximum stack depth per invocation:  3 words
 4537
 4538
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (25)

;          3186  routine WRT_MR (TST_PAT, index) : novalue =
;          3187      begin
;          3188
;          3189  !++
;          3190  ! FUNCTIONAL DESCRIPTION:
;          3191  !
;          3192  !       LOADS THE MAINTENANCE REGISTER WITH A DATA PATTERN GENERATED BY THE MACRO WRT_MASK
;          3193  !
;          3194  ! FORMAL PARAMETERS:
;          3195  !
;          3196  !       TST_PAT
;          3197  !       CURRENT DATA PATTERN TO BE LOADED IN THE REGISTER.
;          3198  !
;          3199  !       INDEX
;          3200  !       USED BY THE MACRO WRT_MASK TO SELECT THE CURRENT REGISTERS ADDRESS,
;          3201  !       FORCED HI, FORCED LO AND DON'T CARE MASK INFORMATION.
;          3202  !--
;          3203
;          3204      MLMR = WRT_MASK;                          !LOAD MLMR WITH GENERATED WRT_MASK PATTERN
;          3205      end;

                                    .SBTTL  WRT.MR
020202  004167  163636        WRT.MR: JSR     R1,$SAVE2                       ;                       3186
020206  016600  000010                MOV     10(SP),R0                       ; INDEX,*               3204
020212  006300                        ASL     R0
020214  006300                        ASL     R0
020216  006300                        ASL     R0
020220  010001                        MOV     R0,R1
020222  016100  013156                MOV     ML.REG+2(R1),R0
020226  056600  000012                BIS     12(SP),R0                       ; TST.PAT,*
020232  046100  013160                BIC     ML.REG+4(R1),R0
020236  016102  013162                MOV     ML.REG+6(R1),R2
020242  050002                        BIS     R0,R2
020244  010277  173024                MOV     R2,@ML.REG+120
020250  000207                        RTS     PC                              ;                       3186

                        ; Routine Size:  20 words
                        ; Maximum stack depth per invocation:  3 words
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 15:05:08    PA:<NEALE>ML4.BLI.7 (26)

;          3206  routine RD_MR (TST_PAT, index, ERR_FLG) : novalue =
;          3207      begin
;          3208
;          3209  !++
;          3210  ! FUNCTIONAL DESCRIPTION:
;          3211  !
;          3212  !       COMPARES THE CONTENTS OF THE
;          3213  !       MAINTENANCE REGISTER WITH THE
;          3214  !       MASKED DATA PATTERN
;          3215  !       GENERATED BY THE MACRO 'WRT_MASK'.
;          3216  !
;          3217  !       IF THE COMPARE IS NOT EQUAL THEN
;          3218  !       THE FORMAL PARAMETER 'ERR_FLG' IS
;          3219  !       ASSIGNED A ONE TO INDICATE THE
;          3220  !       ERROR.
;          3221  !
;          3222  ! FORMAL PARAMETERS:
;          3223  !
;          3224  !       TST_PAT
;          3225  !       DATA PATTERN TO BE MASKED AND
;          3226  !       COMPARED AGAINST THE CONTENTS
;          3227  !       OF THE REGISTER UNDER TEST.
;          3228  !
;          3229  !       INDEX
;          3230  !       USED BY THE MACRO WRT_MASK TO
;          3231  !       SELECT THE CURRENT REGISTERS ADDRESS,
;          3232  !       FORCED HI, FORCED LO AND DON'T CARE
;          3233  !       MASK INFORMATION.
;          3234  !
;          3235  !       ERR_FLG
;          3236  !       CONTAINS THE ADDRESS (PASSED BY REF)
;          3237  !       OF THE CALLERS ERROR FLG TO ENABLE THE
;          3238  !       CALLER TO EXAMINE THE ERROR STATUS
;          3239  !       OF THE ROUTINE CALL.
;          3240  !
;          3241  ! IMPLICIT INPUTS:
;          3242  !       WT_DATA
;          3243  !         GETS LOADED WITH THE GENERATED
;          3244  !         WRT_MASK DATA PATTERN THUS ALLOWING
;          3245  !         CALLER TO PRINT FAILING GOOD DATA.
;          3246  !
;          3247  !       RD_DATA
;          3248  !         GETS LOADED WITH DATA READ FROM THE
;          3249  !         REGISTER THUS ALLOWING CALLER
;          3250  !         TO PRINT FAILING BAD DATA.
;          3251  !
;          3252  ! IMPLICIT OUTPUTS:
;          3253  !       GLOBAL LOCATION WR_DATA
;          3254  !       AND RD_DATA LOADED WITH GOOD
;          3255  !       AND BAD REGISTER DATA
;          3256  !
;          3257  !--
```

```
4650 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
4651 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (26)
4652
4653 ;          3258
4654 ;          3259      .ERR_FLG = ZERO;                   !CLEAR THE ERROR FLAG
4655 ;          3260      WT_DATA = WRT_MASK;                !SAVE THE DATA WRITTEN TO THE REGISTER
4656 ;          3261      RD_DATA = .ML.MR or .IGNORE;       !READ AND SAVE THE REGISTER
4657 ;          3262
4658 ;          3263      if .WT_DATA neq .RD_DATA then .ERR_FLG = ONE;       !READ THE REG FOR WRT_MASK
4659 ;          3264
4660 ;          3265                                         !SET ERROR FLAG IF NEQ
4661 ;          3266      end;
4665
4666                                    .SBTTL  RD.MR
4670 020252  004167  163566          RD.MR:  JSR     R1,$SAVE2                       ;                       3206
4671 020256  005076  000010                  CLR     @10(SP)                         ; ERR.FLG               3259
4672 020262  016600  000012                  MOV     12(SP),R0                       ; INDEX,*               3260
4673 020266  006300                          ASL     R0
4674 020270  006300                          ASL     R0
4675 020272  006300                          ASL     R0
4676 020274  010001                          MOV     R0,R1
4677 020276  016100  013156                  MOV     ML.REG+2(R1),R0
4678 020302  056600  000014                  BIS     14(SP),R0                       ; 1ST.PAT,*
4679 020306  046100  013160                  BIC     ML.REG+4(R1),R0
4680 020312  016102  013162                  MOV     ML.REG+6(R1),R2
4681 020316  050002                          BIS     R0,R2
4682 020320  010267  172602                  MOV     R2,WT.DATA
4683 020324  017702  172744                  MOV     @ML.REG+120,R2                  ;                       3261
4684 020330  056102  013162                  BIS     ML.REG+6(R1),R2
4685 020334  010267  172570                  MOV     R2,RD.DATA
4686 020340  026767  172562  172562          CMP     WT.DATA,RD.DATA                 ;                       3263
4687 020346  001403                          BEQ     1$
4688 020350  012776  000001  000010          MOV     #1,@10(SP)                      ; *,ERR.FLG
4689 020356  000207                  1$:     RTS     PC                              ;                       3206
4690
4691                                    ; Routine Size:  35 words
4692                                    ; Maximum stack depth per invocation:  3 words
4697
4698
```

```
4700 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
4701 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (27)
4702
4703 ;          3267  routine WRT_PA (TST_PAT, index) : novalue =
4704 ;          3268      begin
4705 ;          3269
4706 ;          3270  !++
4707 ;          3271  ! FUNCTIONAL DESCRIPTION:
4708 ;          3272  !       LOADS THE PROM ADDRESS REGISTER WITH A DATA PATTERN GENERATED BY THE MACRO WRT_MASK
4709 ;          3273  ! FORMAL PARAMETERS:
4710 ;          3274  !       TST_PAT
4711 ;          3275  !       CURRENT DATA PATTERN TO BE LOADED IN THE REGISTER.
4712 ;          3276  !       INDEX
4713 ;          3277  !       USED BY THE MACRO WRT_MASK TO SELECT THE CURRENT REGISTERS ADDRESS,
4714 ;          3278  !       FORCED HI, FORCED LO AND DON'T CARE MASK INFORMATION.
4715 ;          3279  !--
4716 ;          3280
4717 ;          3281      PROM_DIS = ONE;                    !SET PROM DISABLE BIT
4718 ;          3282      MLPA = WRT_MASK;                   !LOAD MLPA WITH GENERATED WRT_MASK PATTERN
4719 ;          3283      PROM_DIS = ZERO;                   !CLEAR PROM DISABLE BIT
4720 ;          3284      end;
4724
4725                                        .SBTTL  WRT.PA
4729 020360  004167  163460          WRT.PA: JSR     R1,$SAVE2                       ;                       3267
4730 020364  152777  000040  172702          BISB    #40,@ML.REG+120                 ;                       3281
4731 020372  016600  000010                  MOV     10(SP),R0                       ; INDEX,*               3282
4732 020376  006300                          ASL     R0
4733 020400  006300                          ASL     R0
4734 020402  006300                          ASL     R0
4735 020404  010001                          MOV     R0,R1
4736 020406  016100  013156                  MOV     ML.REG+2(R1),R0
4737 020412  056600  000012                  BIS     12(SP),R0                       ; TST.PAT,*
4738 020416  046100  013160                  BIC     ML.REG+4(R1),R0
4739 020422  016102  013162                  MOV     ML.REG+6(R1),R2
4740 020426  050002                          BIS     R0,R2
4741 020430  010277  172620                  MOV     R2,@ML.REG+100
4742 020434  142777  000040  172632          BICB    #40,@ML.REG+120                 ;                       3283
4743 020442  000207                          RTS     PC                              ;                       3267
4744
4745                                ; Routine Size:  26 words
4746                                ; Maximum stack depth per invocation:  3 words
4751
4752
```

```
4754 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
4755 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (28)
4756
4757 ;          3285  routine RD_PA (TST_PAT, index, ERR_FLG) : novalue =
4758 ;          3286      begin
4759 ;          3287
4760 ;          3288  !++
4761 ;          3289  ! FUNCTIONAL DESCRIPTION:
4762 ;          3290  !
4763 ;          3291  !       COMPARES THE CONTENTS OF THE
4764 ;          3292  !       PROM ADDRESS REGISTER
4765 ;          3293  !       WITH THE MASKED DATA PATTERN
4766 ;          3294  !       GENERATED BY THE MACRO 'WRT_MASK'.
4767 ;          3295  !
4768 ;          3296  !       IF THE COMPARE IS NOT EQUAL THEN
4769 ;          3297  !       THE FORMAL PARAMETER 'ERR_FLG' IS
4770 ;          3298  !       ASSIGNED A ONE TO INDICATE THE
4771 ;          3299  !       ERROR.
4772 ;          3300  !
4773 ;          3301  ! FORMAL PARAMETERS:
4774 ;          3302  !       TST_PAT
4775 ;          3303  !       DATA PATTERN TO BE MASKED AND
4776 ;          3304  !       COMPARED AGAINST THE CONTENTS
4777 ;          3305  !       OF THE REGISTER UNDER TEST.
4778 ;          3306  !
4779 ;          3307  !       INDEX
4780 ;          3308  !       USED BY THE MACRO WRT_MASK TO
4781 ;          3309  !       SELECT THE CURRENT REGISTERS ADDRESS,
4782 ;          3310  !       FORCED HI, FORCED LO AND DON'T CARE
4783 ;          3311  !       MASK INFORMATION.
4784 ;          3312  !
4785 ;          3313  !       ERR_FLG
4786 ;          3314  !       CONTAINS THE ADDRESS (PASSED BY REF)
4787 ;          3315  !       OF THE CALLERS ERROR FLG TO ENABLE THE
4788 ;          3316  !       CALLER TO EXAMINE THE ERROR STATUS
4789 ;          3317  !       OF THE ROUTINE CALL.
4790 ;          3318  !
4791 ;          3319  ! IMPLICIT INPUTS:
4792 ;          3320  !       WT_DATA
4793 ;          3321  !         GETS LOADED WITH THE GENERATED
4794 ;          3322  !         WRT_MASK DATA PATTERN THUS ALLOWING
4795 ;          3323  !         CALLER TO PRINT FAILING GOOD DATA.
4796 ;          3324  !
4797 ;          3325  !       RD_DATA
4798 ;          3326  !         GETS LOADED WITH DATA READ FROM THE
4799 ;          3327  !         REGISTER THUS ALLOWING CALLER
4800 ;          3328  !         TO PRINT FAILING BAD DATA.
4801 ;          3329  !
4802 ;          3330  ! IMPLICIT OUTPUTS:
4803 ;          3331  !       GLOBAL LOCATION WR_DATA
4804 ;          3332  !       AND RD_DATA LOADED WITH GOOD
4805 ;          3333  !       AND BAD REGISTER DATA
4806 ;          3334  !
4807 ;          3335  !--
4808 ;          3336
```

```
4810 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
4811 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (28)
4812
4813 ;          3337      .ERR_FLG = ZERO;                   !CLEAR THE ERROR FLAG
4814 ;          3338      PROM_DIS = ONE;                    !SET THE PROM DISABLE BIT
4815 ;          3339      WT_DATA = WRT_MASK;                !SAVE THE DATA WRITTEN TO THE REGISTER
4816 ;          3340      RD_DATA = .ML@PA or .IGNORE;       !READ AND SAVE THE REGISTER
4817 ;          3341
4818 ;          3342      if .WT_DATA neq .RD_DATA then .ERR_FLG = ONE;       !READ THE REG FOR WRT_MASK
4819 ;          3343
4820 ;          3344      PROM_DIS = ZERO;                   !SET ERROR IF NEQ
4821 ;          3345      end;                               !CLEAR THE PROM DISABLE BIT
4825
4826                                        .SBTTL  RD.PA
4830 020444  004167  163374          RD.PA:  JSR     R1,$SAVE2                       ;                       3285
4831 020450  005076  000010                  CLR     @10(SP)                         ; ERR.FLG               3337
4832 020454  152777  000040  172612          BISB    #40,@ML.REG+120                 ;                       3338
4833 020462  016600  000012                  MOV     12(SP),R0                       ; INDEX,*               3339
4834 020466  006300                          ASL     R0
4835 020470  006300                          ASL     R0
4836 020472  006300                          ASL     R0
4837 020474  010001                          MOV     R0,R1
4838 020476  016100  013156                  MOV     ML.REG+2(R1),R0
4839 020502  056600  000014                  BIS     14(SP),R0                       ; TST.PAT,*
4840 020506  046100  013160                  BIC     ML.REG+4(R1),R0
4841 020512  016102  013162                  MOV     ML.REG+6(R1),R2
4842 020516  050002                          BIS     R0,R2
4843 020520  010267  172402                  MOV     R2,WT.DATA
4844 020524  017702  172524                  MOV     @ML.REG+100,R2                  ;                       3340
4845 020530  056102  013162                  BIS     ML.REG+6(R1),R2
4846 020534  010267  172370                  MOV     R2,RD.DATA
4847 020540  026767  172362  172362          CMP     WT.DATA,RD.DATA                 ;                       3342
4848 020546  001403                          BEQ     1$
4849 020550  012776  000001  000010          MOV     #1,@10(SP)                      ; *,ERR.FLG
4850 020556  142777  000040  172510  1$:     BICB    #40,@ML.REG+120                 ;                       3344
4851 020564  000207                          RTS     PC                              ;                       3285
4852
4853                                ; Routine Size:  41 words
4854                                ; Maximum stack depth per invocation:  3 words
4859
4860
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
:                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (29)

routine WRT_E1 (TST_PAT, index) : novalue =
    begin

!++
! FUNCTIONAL DESCRIPTION:
!       LOADS THE ECC CRC WORD REG 1 WITH A DATA PATTERN GENERATED BY THE MACRO WRT_MASK
! FORMAL PARAMETERS:
!       TST_PAT
!       CURRENT DATA PATTERN TO BE LOADED IN THE REGISTER.
!       INDEX
!       USED BY THE MACRO WRT_MASK TO SELECT THE CURRENT REGISTERS ADDRESS,
!       FORCED HI, FORCED LO AND DON'T CARE MASK INFORMATION.
!--

    ECC_DM = ONE;                               !SET ECC DIAG MODE
    MLE1 = WRT_MASK;                            !LOAD MLE1 WITH GENERATED WRT_MASK PATTERN
    ECC_DM = ZERO;                              !CLEAR ECC DIAG MODE
    end;

                                        .SBTTL  WRT.E1
020566  004167  163252          WRT.E1: JSR     R1,$SAVE2                       ;                       3346
020572  152777  000001  172474          BISB    #1,@ML.REG+120                  ;                       3360
020600  016600  000010                  MOV     10(SP),R0                       ; INDEX,*               3361
020604  006300                          ASL     R0
020606  006300                          ASL     R0
020610  006300                          ASL     R0
020612  010001                          MOV     R0,R1
020614  016100  013156                  MOV     ML.REG+2(R1),R0
020620  056600  000012                  BIS     12(SP),R0                       ; TST.PAT,*
020624  046100  013160                  BIC     ML.REG+4(R1),R0
020630  016102  013162                  MOV     ML.REG+6(R1),R2
020634  050002                          BIS     R0,R2
020636  010277  172462                  MOV     R2,@ML.REG+150
020642  142777  000001  172424          BICB    #1,@ML.REG+120                  ;                       3362
020650  000207                          RTS     PC                              ;                       3346

                                ; Routine Size:  26 words
                                ; Maximum stack depth per invocation:  3 words
```

```
 4916 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
 4917 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (30)
 4918
 4919 ;          3364  routine RD_E1 (TST_PAT, index, ERR_FLG) : novalue =
 4920 ;          3365      begin
 4921 ;          3366
 4922 ;          3367  !++
 4923 ;          3368  ! FUNCTIONAL DESCRIPTION:
 4924 ;          3369  !
 4925 ;          3370  !       COMPARES THE CONTENTS OF THE
 4926 ;          3371  !       ECC CRC WORD REGISTER 1
 4927 ;          3372  !       WITH THE MASKED DATA PATTERN
 4928 ;          3373  !       GENERATED BY THE MACRO 'WRT_MASK'
 4929 ;          3374  !
 4930 ;          3375  !       IF THE COMPARE IS NOT EQUAL THEN
 4931 ;          3376  !        THE FORMAL PARAMETER 'ERR_FLG' IS
 4932 ;          3377  !       ASSIGNED A ONE TO INDICATE THE ERROR.
 4933 ;          3378  !
 4934 ;          3379  ! FORMAL PARAMETERS:
 4935 ;          3380  !       TST_PAT
 4936 ;          3381  !       DATA PATTERN TO BE MASKED AND
 4937 ;          3382  !       COMPARED AGAINST THE CONTENTS
 4938 ;          3383  !       OF THE REGISTER UNDER TEST.
 4939 ;          3384  !
 4940 ;          3385  !       INDEX
 4941 ;          3386  !       USED BY THE MACRO WRT_MASK TO
 4942 ;          3387  !       SELECT THE CURRENT REGISTERS ADDRESS,
 4943 ;          3388  !       FORCED HI, FORCED LO AND DON'T CARE
 4944 ;          3389  !       MASK INFORMATION.
 4945 ;          3390  !
 4946 ;          3391  !       ERR_FLG
 4947 ;          3392  !       CONTAINS THE ADDRESS (PASSED BY REF)
 4948 ;          3393  !       OF THE CALLERS ERROR FLG TO ENABLE THE
 4949 ;          3394  !       CALLER TO EXAMINE THE ERROR STATUS
 4950 ;          3395  !       OF THE ROUTINE CALL.
 4951 ;          3396  !
 4952 ;          3397  ! IMPLICIT INPUTS:
 4953 ;          3398  !       WT_DATA
 4954 ;          3399  !         GETS LOADED WITH THE GENERATED
 4955 ;          3400  !         WRT_MASK DATA PATTERN THUS ALLOWING
 4956 ;          3401  !         CALLER TO PRINT FAILING GOOD DATA.
 4957 ;          3402  !
 4958 ;          3403  !       RD_DATA
 4959 ;          3404  !         GETS LOADED WITH DATA READ FROM THE
 4960 ;          3405  !         REGISTER THUS ALLOWING CALLER
 4961 ;          3406  !         TO PRINT FAILING BAD DATA.
 4962 ;          3407  !
 4963 ;          3408  ! IMPLICIT OUTPUTS:
 4964 ;          3409  !       GLOBAL LOCATION WR_DATA
 4965 ;          3410  !       AND RD_DATA LOADED WITH GOOD
 4966 ;          3411  !       AND BAD REGISTER DATA
 4967 ;          3412  !
 4968 ;          3413  !--
 4969 ;          3414
 4970 ;          3415      .ERR_FLG = ZERO;                          !CLEAR THE ERROR FLAG
```

```
 4972 ;ML4                                                          14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
 4973 ;                                                             14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (30)
 4974
 4975 ;          3416      ECC_DM = ONE;                            !SET ECC DIAG MODE
 4976 ;          3417      WT_DATA = WRT_MASK;                      !SAVE THE DATA WRITTEN TO THE REGISTER
 4977 ;          3418      RD_DATA = .MLE1 or .IGNORE;              !READ AND SAVE THE REGISTER
 4978 ;          3419
 4979 ;          3420      if .WT_DATA neq .RD_DATA then .ERR_FLG = ONE;       !READ THE REG FOR WRT_MASK
 4980 ;          3421
 4981 ;          3422      ECC_DM = ZERO;                           !SET ERROR FLAG IF NEQ
 4982 ;          3423                                               !CLEAR ECC DIAG MODE
 4983 ;          3424      end;
 4987
 4988                                        .SBTTL  RD.E1
 4992 020652  004167  163166          RD.E1: JSR     R1,$SAVE2                ;                                    3364
 4993 020656  005076  000010                 CLR     @10(SP)                  ; ERR.FLG                          3415
 4994 020662  152777  000001  172404         BISB    #1,@ML.REG+120           ;                                  3416
 4995 020670  016600  000012                 MOV     12(SP),R0                ; INDEX,*                          3417
 4996 020674  006300                         ASL     R0
 4997 020676  006300                         ASL     R0
 4998 020700  006300                         ASL     R0
 4999 020702  010001                         MOV     R0,R1
 5000 020704  016100  013156                 MOV     ML.REG+2(R1),R0
 5001 020710  056600  000014                 BIS     14(SP),R0                ; TST.PAT,*
 5002 020714  046100  013160                 BIC     ML.REG+4(R1),R0
 5003 020720  016102  013162                 MOV     ML.REG+6(R1),R2
 5004 020724  050002                         BIS     R0,R2
 5005 020726  010267  172174                 MOV     R2,WT.DATA
 5006 020732  017702  172366                 MOV     @ML.REG+150,R2           ;                                  3418
 5007 020736  056102  013162                 BIS     ML.REG+6(R1),R2
 5008 020742  010267  172162                 MOV     R2,RD.DATA
 5009 020746  026767  172154  172154         CMP     WT.DATA,RD.DATA          ;                                  3420
 5010 020754  001403                         BEQ     1$
 5011 020756  012776  000001  000010         MOV     #1,@10(SP)               ; *,ERR.FLG
 5012 020764  142777  000001  172302  1$:    BICB    #1,@ML.REG+120           ;                                  3422
 5013 020772  000207                         RTS     PC                       ;                                  3364
 5014
 5015                                ; Routine Size:  41 words
 5016                                ; Maximum stack depth per invocation:  3 words
 5021
 5022
```

```
5024 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
5025 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (31)
5026
5027 ;          3425  routine WRT_E2 (TST_PAT, index) : novalue =
5028 ;          3426      begin
5029 ;          3427
5030 ;          3428  !++
5031 ;          3429  ! FUNCTIONAL DESCRIPTION:
5032 ;          3430  !
5033 ;          3431  !       LOADS THE ECC CRC WORD
5034 ;          3432  !       REGISTER 2 WITH A DATA PATTERN
5035 ;          3433  !       GENERATED BY THE MACRO
5036 ;          3434  !       WRT_MASK
5037 ;          3435  !
5038 ;          3436  ! FORMAL PARAMETERS:
5039 ;          3437  !
5040 ;          3438  !       TST_PAT
5041 ;          3439  !       CURRENT DATA PATTERN TO BE LOADED IN THE REGISTER.
5042 ;          3440  !
5043 ;          3441  !       INDEX
5044 ;          3442  !       USED BY THE MACRO WRT_MASK
5045 ;          3443  !       TO SELECT THE CURRENT REGISTERS
5046 ;          3444  !       ADDRESS, FORCED HI, FORCED
5047 ;          3445  !       LO AND DON'T CARE MASK
5048 ;          3446  !       INFORMATION.
5049 ;          3447  !
5050 ;          3448  !--
5051 ;          3449
5052 ;          3450      ECC_DM = ONE;                          !SET ECC DIAG MODE
5053 ;          3451      MLE2_MASK = %o'177700';                !MASK OUT DATA DIAG BITS
5054 ;          3452      MLE2 = WRT_MASK;                       !LOAD MLE2 WITH GENERATED WRT_MASK PATTERN
5055 ;          3453      MLE2_MASK = %o'000000';                !RESTORE MLE2_MASK
5056 ;          3454      ECC_DM = ZERO;                         !CLEAR ECC DIAG MODE
5057 ;          3455      end;
5061
5062                                        .SBTTL  WRT.E2
5066 020774  004167  163044          WRT.E2: JSR     R1,$SAVE2                       ;                       3425
5067 021000  152777  000001  172266          BISB    #1,@ML.REG+120                  ;                       3450
5068 021006  012767  177700  172326          MOV     #-100,ML.REG+166                ;                       3451
5069 021014  016600  000010                  MOV     10(SP),R0                       ; INDEX,*               3452
5070 021020  006300                          ASL     R0
5071 021022  006300                          ASL     R0
5072 021024  006300                          ASL     R0
5073 021026  010001                          MOV     R0,R1
5074 021030  016100  013156                  MOV     ML.REG+2(R1),R0
5075 021034  056600  000012                  BIS     12(SP),R0                       ; TST.PAT,*
5076 021040  046100  013160                  BIC     ML.REG+4(R1),R0
5077 021044  016102  013162                  MOV     ML.REG+6(R1),R2
5078 021050  050002                          BIS     R0,R2
```

```
5080                            ;ML4                                                    14-May-1981 13:48:54    TOPS
5081                            ;                                                       14-May-1981 13:05:08    PA:<
5082
5083 021052  010277  172256                MOV     R2,@ML.REG+160
5084 021056  005067  172260                CLR     ML.REG+166                  ;                           3453
5085 021062  142777  000001  172204        BICB    #1,@ML.REG+120              ;                           3454
5086 021070  000207                        RTS     PC                          ;                           3425
5087
5088                            ; Routine Size:  31 words
5089                            ; Maximum stack depth per invocation:  3 words
5094
5095
```

```
5097 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
5098 :                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (32)
5099
5100 ;          3456  routine RD_E2 (TST_PAT, index, ERR_FLG) : novalue =
5101 ;          3457      begin
5102 ;          3458
5103 ;          3459  !++
5104 ;          3460  ! FUNCTIONAL DESCRIPTION:
5105 ;          3461  !
5106 ;          3462  !       COMPARES THE CONTENTS OF THE
5107 ;          3463  !       ECC CRC WORD REGISTER 2
5108 ;          3464  !       WITH THE MASKED DATA PATTERN
5109 ;          3465  !       GENERATED BY THE MACRO 'WRT_MASK'
5110 ;          3466  !
5111 ;          3467  !       IF THE COMPARE IS NOT EQUAL THEN
5112 ;          3468  !       THE FORMAL PARAMETER 'ERR_FLG' IS
5113 ;          3469  !       ASSIGNED A ONE TO INDICATE THE
5114 ;          3470  !       ERROR.
5115 ;          3471  !
5116 ;          3472  ! FORMAL PARAMETERS:
5117 ;          3473  !
5118 ;          3474  !       TST_PAT
5119 ;          3475  !       DATA PATTERN TO BE MASKED AND
5120 ;          3476  !       COMPARED AGAINST THE CONTENTS
5121 ;          3477  !       OF THE REGISTER UNDER TEST.
5122 ;          3478  !
5123 ;          3479  !       INDEX
5124 ;          3480  !       USED BY THE MACRO WRT_MASK TO
5125 ;          3481  !       SELECT THE CURRENT REGISTERS ADDRESS,
5126 ;          3482  !       FORCED HI, FORCED LO AND DON'T CARE
5127 ;          3483  !       MASK INFORMATION.
5128 ;          3484  !
5129 ;          3485  !       ERR_FLG
5130 ;          3486  !       CONTAINS THE ADDRESS (PASSED BY REF)
5131 ;          3487  !       OF THE CALLERS ERROR FLG TO ENABLE THE
5132 ;          3488  !       CALLER TO EXAMINE THE ERROR STATUS
5133 ;          3489  !       OF THE ROUTINE CALL.
5134 ;          3490  !
5135 ;          3491  ! IMPLICIT INPUTS:
5136 ;          3492  !       WT_DATA
5137 ;          3493  !         GETS LOADED WITH THE GENERATED
5138 ;          3494  !         WRT_MASK DATA PATTERN THUS ALLOWING
5139 ;          3495  !         CALLER TO PRINT FAILING GOOD DATA.
5140 ;          3496  !
5141 ;          3497  !       RD_DATA
5142 ;          3498  !         GETS LOADED WITH DATA READ FROM THE
5143 ;          3499  !         REGISTER THUS ALLOWING CALLER
5144 ;          3500  !         TO PRINT FAILING BAD DATA.
5145 ;          3501  !
5146 ;          3502  ! IMPLICIT OUTPUTS:
5147 ;          3503  !       GLOBAL LOCATION WR_DATA
5148 ;          3504  !       AND RD_DATA LOADED WITH GOOD
5149 ;          3505  !       AND BAD REGISTER DATA
5150 ;          3506  !
5151 ;          3507  !--
```

```
5153 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
5154 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (32)
5155
5156 ;          3508
5157 ;          3509        .ERR_FLG = ZERO;                 !CLEAR ERROR FLAG
5158 ;          3510        ECC_DM = ONE;                    !SET ECC DIAG MODE
5159 ;          3511        MLE2_MASK = %o'177700';          !MASK OUT DATA DIAG BITS
5160 ;          3512        WT_DATA = WRT_MASK;              !SAVE THE DATA WRITTEN TO THE REGISTER
5161 ;          3513        RD_DATA = .MLE2 or .IGNORE;      !READ AND SAVE THE REGISTER
5162 ;          3514
5163 ;          3515        if .WT_DATA neq .RD_DATA then .ERR_FLG = ONE;       !READ THE REG FOR WRT_MASK
5164 ;          3516
5165 ;          3517                                         !SET ERROR FLG IF NEQ
5166 ;          3518        MLE2_MASK = %o'000000';          !RESTORE MLE2 MASK
5167 ;          3519        ECC_DM = ZERO;                   !CLEAR ECC DIAG MODE
5168 ;          3520        end;
5172
5173                                          .SBTTL  RD.E2
5177 021072  004167  162746          RD.E2:  JSR     R1,$SAVE2                       ;                       3456
5178 021076  005076  000010                  CLR     @10(SP)                         ; ERR.FLG               3509
5179 021102  152777  000001  172164          BISB    #1,@ML.REG+120                  ;                       3510
5180 021110  012767  177700  172224          MOV     #-100,ML.REG+166                ;                       3511
5181 021116  016600  000012                  MOV     12(SP),R0                       ; INDEX,*               3512
5182 021122  006300                          ASL     R0
5183 021124  006300                          ASL     R0
5184 021126  006300                          ASL     R0
5185 021130  010001                          MOV     R0,R1
5186 021132  016100  013156                  MOV     ML.REG+2(R1),R0
5187 021136  056600  000014                  BIS     14(SP),R0                       ; TST.PAT,*
5188 021142  046100  013160                  BIC     ML.REG+4(R1),R0
5189 021146  016102  013162                  MOV     ML.REG+6(R1),R2
5190 021152  050002                          BIS     R0,R2
5191 021154  010267  171746                  MOV     R2,WT.DATA
5192 021160  017702  172150                  MOV     @ML.REG+160,R2                  ;                       3513
5193 021164  056102  013162                  BIS     ML.REG+6(R1),R2
5194 021170  010267  171734                  MOV     R2,RD.DATA
5195 021174  026767  171726  171726          CMP     WT.DATA,RD.DATA                 ;                       3515
5196 021202  001403                          BEQ     1$
5197 021204  012776  000001  000010          MOV     #1,@10(SP)                      ; *,ERR.FLG
5198 021212  005067  172124          1$:     CLR     ML.REG+166                      ;                       3518
5199 021216  142777  000001  172050          BICB    #1,@ML.REG+120                  ;                       3519
5200 021224  000207                          RTS     PC                              ;                       3456
5201
5202                                 ; Routine Size:  46 words
5203                                 ; Maximum stack depth per invocation:  3 words
```

```
5209 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
5210 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (33)
5211
5212 ;          3521  routine WRT_PD (TST_PAT, index) : novalue =
5213 ;          3522      begin
5214 ;          3523
5215 ;          3524  !++
5216 ;          3525  ! FUNCTIONAL DESCRIPTION:
5217 ;          3526  !
5218 ;          3527  !       LOADS THE PROM DATA
5219 ;          3528  !       REGISTER WITH A DATA PATTERN
5220 ;          3529  !       GENERATED BY THE MACRO
5221 ;          3530  !       WRT_MASK
5222 ;          3531  !
5223 ;          3532  ! FORMAL PARAMETERS:
5224 ;          3533  !
5225 ;          3534  !       TST_PAT
5226 ;          3535  !       CURRENT DATA PATTERN TO BE
5227 ;          3536  !       LOADED IN THE REGISTER.
5228 ;          3537  !
5229 ;          3538  !       INDEX
5230 ;          3539  !       USED BY THE MACRO WRT_MASK
5231 ;          3540  !       TO SELECT THE CURRENT REGISTERS
5232 ;          3541  !       ADDRESS, FORCED HI, FORCED
5233 ;          3542  !       LO AND DON'T CARE MASK
5234 ;          3543  !       INFORMATION
5235 ;          3544  !
5236 ;          3545  !--
5237 ;          3546
5238 ;          3547      PROM_RW = ONE;                     !SET PROM READ WRITE
5239 ;          3548      DAT_DM = ONE;                      !SET DATA DIAG MODE
5240 ;          3549      MLPD = WRT_MASK;                   !LOAD MLPD WITH GENERATED WRT_MASK PATTERN
5241 ;          3550      DAT_CLK = ONE;                     !DO A DATA CLK
5242 ;          3551      PROM_RW = ZERO;                    !CLEAR PROM READ WRITE
5243 ;          3552      DAT_DM = ZERO;                     !CLEAR DATA DIAG MODE
5244 ;          3553      end;
5248
5249                                        .SBTTL  WRT.PD
5253 021226  004167  162612          WRT.PD: JSR     R1,$SAVE2                  ;                    3521
5254 021232  152777  000100  172034          BISB    #100,@ML.REG+120           ;                    3547
5255 021240  152777  000010  172026          BISB    #10,@ML.REG+120            ;                    3548
5256 021246  016600  000010                  MOV     10(SP),R0                  ; INDEX,*            3549
5257 021252  006300                          ASL     R0
5258 021254  006300                          ASL     R0
5259 021256  006300                          ASL     R0
5260 021260  010001                          MOV     R0,R1
5261 021262  016100  013156                  MOV     ML.REG+2(R1),R0
5262 021266  056600  000012                  BIS     12(SP),R0                  ; TST.PAT,*
5263 021272  046100  013160                  BIC     ML.REG+4(R1),R0
```

```
5265                                    ;ML4                                                    14-May-1981 13:48:54    TOPS
5266                                    ;                                                       14-May-1981 13:05:08    PA:<
5267
5268 021276  016102  013162                       MOV     ML.REG+6(R1),R2
5269 021302  050002                              BIS     R0,R2
5270 021304  010277  172074                       MOV     R2,@ML.REG+230
5271 021310  152777  000020  171756               BISB    #20,@ML.REG+120                 ;       3550
5272 021316  142777  000100  171750               BICB    #100,@ML.REG+120                ;       3551
5273 021324  142777  000010  171742               BICB    #10,@ML.REG+120                 ;       3552
5274 021332  000207                              RTS     PC                              ;       3521
5275
5276                                    ; Routine Size:  35 words
5277                                    ; Maximum stack depth per invocation:  3 words
5282
5283
```

```
5285 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
5286 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (34)
5287
5288 ;          3554  routine RD_PD (TST_PAT, index, ERR_FLG) : novalue =
5289 ;          3555      begin
5290 ;          3556
5291 ;          3557  !++
5292 ;          3558  ! FUNCTIONAL DESCRIPTION:
5293 ;          3559  !
5294 ;          3560  !       COMPARES THE CONTENTS OF THE
5295 ;          3561  !       PROM DATA REGISTER
5296 ;          3562  !       WITH THE MASKED DATA PATTERN
5297 ;          3563  !       GENERATED BY THE MACRO 'WRT_MASK'
5298 ;          3564  !
5299 ;          3565  !       IF THE COMPARE IS NOT EQUAL THEN
5300 ;          3566  !       THE FORMAL PARAMETER 'ERR FLG' IS
5301 ;          3567  !       ASSIGNED A ONE TO INDICATE THE
5302 ;          3568  !       ERROR
5303 ;          3569  !
5304 ;          3570  ! FORMAL PARAMETERS:
5305 ;          3571  !
5306 ;          3572  !       TST_PAT
5307 ;          3573  !       DATA PATTERN TO BE MASKED AND
5308 ;          3574  !       COMPARED AGAINST THE CONTENTS
5309 ;          3575  !       OF THE REGISTER UNDER TEST
5310 ;          3576  !
5311 ;          3577  !       INDEX
5312 ;          3578  !       USED BY THE MACRO WRT_MASK TO
5313 ;          3579  !       SELECT THE CURRENT REGISTERS ADDRESS,
5314 ;          3580  !       FORCED HI, FORCED LO AND DON'T CARE
5315 ;          3581  !       MASK INFORMATION.
5316 ;          3582  !
5317 ;          3583  !       ERR_FLG
5318 ;          3584  !       CONTAINS THE ADDRESS (PASSED BY REF)
5319 ;          3585  !       OF THE CALLERS ERROR FLG TO ENABLE THE
5320 ;          3586  !       CALLER TO EXAMINE THE ERROR STATUS
5321 ;          3587  !       OF THE ROUTINE CALL.
5322 ;          3588  !
5323 ;          3589  ! IMPLICIT INPUTS:
5324 ;          3590  !       WT_DATA
5325 ;          3591  !         GETS LOADED WITH THE GENERATED
5326 ;          3592  !         WRT_MASK DATA PATTERN THUS ALLOWING
5327 ;          3593  !         CALLER TO PRINT FAILING GOOD DATA.
5328 ;          3594  !
5329 ;          3595  !       RD_DATA
5330 ;          3596  !         GETS LOADED WITH DATA READ FROM THE
5331 ;          3597  !         REGISTER THUS ALLOWING CALLER
5332 ;          3598  !         TO PRINT FAILING BAD DATA.
5333 ;          3599  !
5334 ;          3600  ! IMPLICIT OUTPUTS:
5335 ;          3601  !       GLOBAL LOCATION WR_DATA
5336 ;          3602  !       AND RD_DATA LOADED WITH GOOD
5337 ;          3603  !       AND BAD REGISTER DATA
5338 ;          3604  !
5339 ;          3605  !--
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (34)

;          3606
;          3607       .ERR_FLG = ZERO;                          !CLEAR ERROR FLG
;          3608       PROM_RW = ONE;                            !SET PROM READ WRITE
;          3609       DAT_DM = ONE;                             !SET DATA DIAG MODE
;          3610       WT_DATA = WRT_MASK;                       !SAVE THE DATA WRITTEN TO THE REGISTER
;          3611       RD_DATA = .MLPD or .IGNORE;               !READ AND SAVE THE REGISTER
;          3612
;          3613       if .WT_DATA neq .RD_DATA then .ERR_FLG = ONE;       !READ THE REG FOR WRT_MASK
;          3614
;          3615                                                 !SET ERROR FLAG IF NEQ
;          3616       PROM_RW = ZERO;                           !CLEAR PROM READ WRTE
;          3617       DAT_DM = ZERO;                            !CLEAR DATA DIAG MODE
;          3618       end;

                                    .SBTTL  RD.PD
021334  004167  162504          RD.PD:  JSR     R1,$SAVE2                       ;                       3554
021340  005076  000010                  CLR     @10(SP)                         ; ERR.FLG               3607
021344  152777  000100  171722          BISB    #100,@ML.REG+120                ;                       3608
021352  152777  000010  171714          BISB    #10,@ML.REG+120                 ;                       3609
021360  016600  000012                  MOV     12(SP),R0                       ; INDEX,*               3610
021364  006300                          ASL     R0
021366  006300                          ASL     R0
021370  006300                          ASL     R0
021372  010001                          MOV     R0,R1
021374  016100  013156                  MOV     ML.REG+2(R1),R0
021400  056600  000014                  BIS     14(SP),R0                       ; TST.PAT,*
021404  046100  013160                  BIC     ML.REG+4(R1),R0
021410  016102  013162                  MOV     ML.REG+6(R1),R2
021414  050002                          BIS     R0,R2
021416  010267  171504                  MOV     R2,WT.DATA
021422  017702  171756                  MOV     @ML.REG+230,R2                  ;                       3611
021426  056102  013162                  BIS     ML.REG+6(R1),R2
021432  010267  171472                  MOV     R2,RD.DATA
021436  026767  171464  171464          CMP     WT.DATA,RD.DATA                 ;                       3613
021444  001403                          BEQ     1$
021446  012776  000001  000010          MOV     #1,@10(SP)                      ; *,ERR.FLG
021454  142777  000100  171612  1$:     BICB    #100,@ML.REG+120                ;                       3616
021462  142777  000010  171604          BICB    #10,@ML.REG+120                 ;                       3617
021470  000207                          RTS     PC                              ;                       3554

                        ; Routine Size:  47 words
                        ; Maximum stack depth per invocation:  3 words
```

```
5397 :ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
5398 :                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (35)
5399
5400 :          3619  routine WRT_EL (TST_PAT, index) : novalue =
5401 :          3620      begin
5402 :          3621
5403 :          3622  !++
5404 :          3623  ! FUNCTIONAL DESCRIPTION:
5405 :          3624  !       DUMMY ROUTINE CALL TO ASSIST IN THE READ
5406 :          3625  !       WRITE REGISTER ALGORITHEM
5407 :          3626  !
5408 :          3627  ! FORMAL PARAMETERS:
5409 :          3628  !       TST_PAT
5410 :          3629  !       DATA PATTERN TO BE MASKED AND
5411 :          3630  !       COMPARED AGAINST THE CONTENTS
5412 :          3631  !       OF THE REGISTER UNDER TEST
5413 :          3632  !
5414 :          3633  !       INDEX
5415 :          3634  !       USED BY THE MACRO WRT_MASK TO
5416 :          3635  !       SELECT THE CURRENT REGISTERS ADDRESS,
5417 :          3636  !       FORCED HI, FORCED LO AND DON'T CARE
5418 :          3637  !       MASK INFORMATION.
5419 :          3638  !
5420 :          3639  !--
5421 :          3640
5422 :          3641  !
5423 :          3642  !  ERROR LOCATION REG IS READ ONLY
5424 :          3643      return;
5425 :          3644      end;
5429
5430                                        .SBTTL  WRT.EL
5434 021472  000207                 WRT.EL: RTS     PC                              ;                    3619
5435
5436                                 ; Routine Size:  1 word
5437                                 ; Maximum stack depth per invocation:  0 words
5442
5443
```

```
5445 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
5446 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (36)
5447
5448 ;          3645  routine RD_EL (TST_PAT, index, ERR_FLG) : novalue =
5449 ;          3646      begin
5450 ;          3647
5451 ;          3648  !++
5452 ;          3649  ! FUNCTIONAL DESCRIPTION:
5453 ;          3650  !
5454 ;          3651  !       COMPARES THE CONTENTS OF THE
5455 ;          3652  !       ERROR LOCATION REGISTER
5456 ;          3653  !       WITH THE CONTENTS OF TST_PAT.
5457 ;          3654  !
5458 ;          3655  !       IF THE COMPARE IS NOT EQUAL THEN
5459 ;          3656  !       THE FORMAL PARAMETER 'ERR FLG' IS
5460 ;          3657  !       ASSIGNED A ONE TO INDICATE THE
5461 ;          3658  !       ERROR
5462 ;          3659  !
5463 ;          3660  ! FORMAL PARAMETERS:
5464 ;          3661  !       TST_PAT
5465 ;          3662  !       DATA PATTERN TO BE MASKED AND
5466 ;          3663  !       COMPARED AGAINST THE CONTENTS
5467 ;          3664  !       OF THE REGISTER UNDER TEST
5468 ;          3665  !
5469 ;          3666  !       INDEX
5470 ;          3667  !       USED BY THE MACRO WRT_MASK TO
5471 ;          3668  !       SELECT THE CURRENT REGISTERS ADDRESS,
5472 ;          3669  !       FORCED HI, FORCED LO AND DON'T CARE
5473 ;          3670  !       MASK INFORMATION.
5474 ;          3671  !
5475 ;          3672  !       ERR_FLG
5476 ;          3673  !       CONTAINS THE ADDRESS (PASSED BY REF)
5477 ;          3674  !       OF THE CALLERS ERROR FLG TO ENABLE THE
5478 ;          3675  !       CALLER TO EXAMINE THE ERROR STATUS
5479 ;          3676  !       OF THE ROUTINE CALL.
5480 ;          3677  !
5481 ;          3678  ! IMPLICIT INPUTS:
5482 ;          3679  !       WT_DATA
5483 ;          3680  !         GETS LOADED WITH THE GENERATED
5484 ;          3681  !         WRT_MASK DATA PATTERN THUS ALLOWING
5485 ;          3682  !         CALLER TO PRINT FAILING GOOD DATA.
5486 ;          3683  !
5487 ;          3684  !       RD_DATA
5488 ;          3685  !         GETS LOADED WITH DATA READ FROM THE
5489 ;          3686  !         REGISTER THUS ALLOWING CALLER
5490 ;          3687  !         TO PRINT FAILING BAD DATA.
5491 ;          3688  !
5492 ;          3689  ! IMPLICIT OUTPUTS:
5493 ;          3690  !       GLOBAL LOCATION WR_DATA
5494 ;          3691  !       AND RD_DATA LOADED WITH GOOD
5495 ;          3692  !       AND BAD REGISTER DATA
5496 ;          3693  !
5497 ;          3694  !--
5498 ;          3695
5499 ;          3696      .ERR_FLG = ZERO;
```

```
5501 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
5502 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (36)
5503
5504 ;          3697      WT_DATA = .TST_PAT;                !SAVE THE DATA WRITTEN TO THE REGISTER
5505 ;          3698      RD_DATA = .MLE[;                   !READ AND SAVE THE REGISTER
5506 ;          3699
5507 ;          3700      if .RD_DATA neq .WT_DATA then .ERR_FLG = ONE;       !READ REGISTER FOR WT_DATA
5508 ;          3701
5509 ;          3702      end;
5513
5514                                        .SBTTL  RD.EL
5518 021474  005076  000002          RD.EL:  CLR     @2(SP)                  ; ERR.FLG                    3696
5519 021500  016667  000006  171420          MOV     6(SP),WT.DATA           ; TST.PAT,*                  3697
5520 021506  017767  171662  171414          MOV     @ML.REG+220,RD.DATA     ;                            3698
5521 021514  026767  171410  171404          CMP     RD.DATA,WT.DATA         ;                            3700
5522 021522  001403                          BEQ     1$
5523 021524  012776  000001  000002          MOV     #1,@2(SP)               ; *,ERR.FLG
5524 021532  000207                    1$:   RTS     PC                      ;                            3645
5525
5526                                ; Routine Size:  16 words
5527                                ; Maximum stack depth per invocation:  0 words
5532
5533
```

```
5535 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
5536 :                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (37)
5537
5538 :          3703  routine WRT_EE (TST_PAT, index) : novalue =
5539 :          3704      begin
5540 :          3705
5541 :          3706  !++
5542 :          3707  ! FUNCTIONAL DESCRIPTION:
5543 :          3708  !       DUMMY ROUTINE CALL TO ASSIST IN THE READ
5544 :          3709  !       WRITE REGISTER ALGORITHEM
5545 :          3710  !
5546 :          3711  ! FORMAL PARAMETERS:
5547 :          3712  !       TST_PAT
5548 :          3713  !       DATA PATTERN TO BE MASKED AND
5549 :          3714  !       COMPARED AGAINST THE CONTENTS
5550 :          3715  !       OF THE REGISTER UNDER TEST
5551 :          3716  !
5552 :          3717  !       INDEX
5553 :          3718  !       USED BY THE MACRO WRT_MASK TO
5554 :          3719  !       SELECT THE CURRENT REGISTERS ADDRESS,
5555 :          3720  !       FORCED HI, FORCED LO AND DON'T CARE
5556 :          3721  !       MASK INFORMATION.
5557 :          3722  !
5558 :          3723  !--
5559 :          3724
5560 :          3725  ! ECC ERROR REGISTER IS READ ONLY
5561 :          3726      return;
5562 :          3727      end;
5566
5567                                    .SBTTL  WRT.EE
5571 021534  000207                WRT.EE: RTS     PC                      ;                    3703
5572
5573                                    ; Routine Size:  1 word
5574                                    ; Maximum stack depth per invocation:  0 words
5579
5580
```

```
5582 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
5583 :                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (38)
5584
5585 ;          3728  routine RD_EE (TST_PAT, index, ERR_FLG) : novalue =
5586 ;          3729      begin
5587 ;          3730
5588 ;          3731  !++
5589 ;          3732  ! FUNCTIONAL DESCRIPTION:
5590 ;          3733  !
5591 ;          3734  !       COMPARES THE CONTENTS OF THE
5592 ;          3735  !       ECC ERROR REGISTER WITH TST_PAT.
5593 ;          3736  !
5594 ;          3737  !       IF THE COMPARE IS NOT EQUAL THEN
5595 ;          3738  !       THE FORMAL PARAMETER 'ERR_FLG' IS
5596 ;          3739  !       ASSIGNED A ONE TO INDICATE THE
5597 ;          3740  !       ERROR.
5598 ;          3741  !
5599 ;          3742  ! FORMAL PARAMETERS:
5600 ;          3743  !       TST_PAT
5601 ;          3744  !       DATA PATTERN TO BE
5602 ;          3745  !       COMPARED AGAINST THE CONTENTS
5603 ;          3746  !       OF THE REGISTER UNDER TEST.
5604 ;          3747  !
5605 ;          3748  !       INDEX
5606 ;          3749  !       USED BY THE MACRO MLEE TO
5607 ;          3750  !       SELECT THE CURRENT REGISTERS ADDRESS.
5608 ;          3751  !
5609 ;          3752  !       ERR_FLG
5610 ;          3753  !       CONTAINS THE ADDRESS (PASSED
5611 ;          3754  !       BY REF) OF THE CALLERS ERROR FLG TO ENABLE
5612 ;          3755  !       THE CALLER TO EXAMINE THE ERROR STATUS
5613 ;          3756  !       OF THE ROUTINE CALL.
5614 ;          3757  !
5615 ;          3758  ! IMPLICIT INPUTS:
5616 ;          3759  !       WT_DATA
5617 ;          3760  !         GETS LOADED WITH THE TST_PAT
5618 ;          3761  !         THUS ALLOWING CALLER TO PRINT
5619 ;          3762  !         THE FAILING DATA.
5620 ;          3763  !
5621 ;          3764  !       RD_DATA
5622 ;          3765  !         GETS LOADED WITH DATA READ FROM THE
5623 ;          3766  !         REGISTER THUS ALLOWING CALLER
5624 ;          3767  !         TO PRINT FAILING BAD DATA.
5625 ;          3768  !
5626 ;          3769  ! IMPLICIT OUTPUTS:
5627 ;          3770  !       GLOBAL LOCATION WR_DATA
5628 ;          3771  !       AND RD_DATA LOADED WITH GOOD
5629 ;          3772  !       AND BAD REGISTER DATA
5630 ;          3773  !
5631 ;          3774  !--
5632 ;          3775
5633 ;          3776      .ERR_FLG = ZERO;                    !CLEAR THE ERROR FLAG
5634 ;          3777      WT_DATA = .TST_PAT;                 !SAVE DATA WRITTEN TO THE REGISTER
5635 ;          3778      RD_DATA = .MLEE;                    !READ AND SAVE THE REGISTER
5636 ;          3779
```

```
5638 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
5639 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (38)
5640
5641 ;          3780      if .RD_DATA neq .WT_DATA then .ERR_FLG = ONE;       !READ MLEE FOR TST_PAT
5642 ;          3781
5643 ;          3782                                                          !SET THE ERROR FLAG IF NEQ
5644 ;          3783      end;
5648
5649                                    .SBTTL  RD.EE
5653 021536  005076  000002          RD.EE:  CLR     @2(SP)                          ; ERR.FLG                    3776
5654 021542  016667  000006  171356          MOV     6(SP),WT.DATA                   ; TST.PAT,*                  3777
5655 021550  017767  171610  171352          MOV     @ML.REG+210,RD.DATA             ;                            3778
5656 021556  026767  171346  171342          CMP     RD.DATA,WT.DATA                 ;                            3780
5657 021564  001403                         BEQ     1$
5658 021566  012776  000001  000002          MOV     #1,@2(SP)                       ; *,ERR.FLG
5659 021574  000207                    1$:   RTS     PC                              ;                            3728
5660
5661                                    ; Routine Size:  16 words
5662                                    ; Maximum stack depth per invocation:  0 words
5667
5668
```

```
5670 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
5671 :                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (39)
5672
5673 ;          3784  routine WRT_D1 (TST_PAT, index) : novalue =
5674 ;          3785      begin
5675 ;          3786
5676 ;          3787  !++
5677 ;          3788  ! FUNCTIONAL DESCRIPTION:
5678 ;          3789  !       LOADS THE DATA DIAG REG 1 WITH A DATA PATTERN GENERATED BY THE MACRO WRT_MASK
5679 ;          3790  ! FORMAL PARAMETERS:
5680 ;          3791  !       TST_PAT
5681 ;          3792  !       CURRENT DATA PATTERN TO BE LOADED IN THE REGISTER.
5682 ;          3793  !       INDEX
5683 ;          3794  !       USED BY THE MACRO WRT_MASK TO SELECT THE CURRENT REGISTERS ADDRESS,
5684 ;          3795  !       FORCED HI, FORCED LO AND DON'T CARE MASK INFORMATION.
5685 ;          3796  !--
5686 ;          3797
5687 ;          3798      DAT_DM = ONE;                          !SET DATA DIAG MODE
5688 ;          3799      MLDT = WRT_MASK;                       !LOAD MLD1 WITH GENERATED WRT_MASK PATTERN
5689 ;          3800      DAT_DM = ZERO;                         !CLEAR DATA DIAG MODE
5690 ;          3801      end;
5694
5695                                        .SBTTL  WRT.D1
5699 021576  004167  162242          WRT.D1: JSR     R1,$SAVE2                       ;                       3784
5700 021602  152777  000010  171464          BISB    #10,@ML.REG+120                 ;                       3798
5701 021610  016600  000010                  MOV     10(SP),R0                       ; INDEX,*               3799
5702 021614  006300                          ASL     R0
5703 021616  006300                          ASL     R0
5704 021620  006300                          ASL     R0
5705 021622  010001                          MOV     R0,R1
5706 021624  016100  013156                  MOV     ML.REG+2(R1),R0
5707 021630  056600  000012                  BIS     12(SP),R0                       ; TST.PAT,*
5708 021634  046100  013160                  BIC     ML.REG+4(R1),R0
5709 021640  016102  013162                  MOV     ML.REG+6(R1),R2
5710 021644  050002                          BIS     R0,R2
5711 021646  010277  171472                  MOV     R2,@ML.REG+170
5712 021652  142777  000010  171414          BICB    #10,@ML.REG+120                 ;                       3800
5713 021660  000207                          RTS     PC                              ;                       3784
5714
5715                                ; Routine Size:  26 words
5716                                ; Maximum stack depth per invocation:  3 words
5721
5722
```

```
5724 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
5725 :                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (40)
5726
5727 :          3802  routine RD_D1 (TST_PAT, index, ERR_FLG) : novalue =
5728 :          3803      begin
5729 :          3804
5730 :          3805  !++
5731 :          3806  ! FUNCTIONAL DESCRIPTION:
5732 :          3807  !
5733 :          3808  !       COMPARES THE CONTENTS OF THE
5734 :          3809  !       DATA DIAG REGISTER 1
5735 :          3810  !       WITH THE MASKED DATA PATTERN
5736 :          3811  !       GENERATED BY THE MACRO 'WRT_MASK''.
5737 :          3812  !
5738 :          3813  !       IF THE COMPARE IS NOT EQUAL THEN
5739 :          3814  !       THE FORMAL PARAMETER 'ERR_FLG'' IS
5740 :          3815  !       ASSIGNED A ONE TO INDICATE THE
5741 :          3816  !       ERROR.
5742 :          3817  !
5743 :          3818  ! FORMAL PARAMETERS:
5744 :          3819  !
5745 :          3820  !       TST_PAT
5746 :          3821  !       DATA PATTERN TO BE MASKED AND
5747 :          3822  !       COMPARED AGAINST THE CONTENTS
5748 :          3823  !       OF THE REGISTER UNDER TEST.
5749 :          3824  !
5750 :          3825  !       INDEX
5751 :          3826  !       USED BY THE MACRO WRT_MASK TO
5752 :          3827  !       SELECT THE CURRENT REGISTERS ADDRESS,
5753 :          3828  !       FORCED HI, FORCED LO AND DON'T CARE
5754 :          3829  !       MASK INFORMATION.
5755 :          3830  !
5756 :          3831  !       ERR_FLG
5757 :          3832  !       CONTAINS THE ADDRESS (PASSED BY REF)
5758 :          3833  !       OF THE CALLERS ERROR FLG TO ENABLE THE
5759 :          3834  !       CALLER TO EXAMINE THE ERROR STATUS
5760 :          3835  !       OF THE ROUTINE CALL.
5761 :          3836  !
5762 :          3837  ! IMPLICIT INPUTS:
5763 :          3838  !       WT_DATA
5764 :          3839  !         GETS LOADED WITH THE GENERATED
5765 :          3840  !         WRT_MASK DATA PATTERN THUS ALLOWING
5766 :          3841  !         CALLER TO PRINT FAILING GOOD DATA.
5767 :          3842  !
5768 :          3843  !       RD_DATA
5769 :          3844  !         GETS LOADED WITH DATA READ FROM THE
5770 :          3845  !         REGISTER THUS ALLOWING CALLER
5771 :          3846  !         TO PRINT FAILING BAD DATA.
5772 :          3847  !
5773 :          3848  ! IMPLICIT OUTPUTS:
5774 :          3849  !       GLOBAL LOCATION WR_DATA
5775 :          3850  !       AND RD_DATA LOADED WITH GOOD
5776 :          3851  !       AND BAD REGISTER DATA
5777 :          3852  !
5778 :          3853  !--
```

```
5780 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
5781 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (40)
5782
5783 ;          3854
5784 ;          3855        .ERR_FLG = ZERO;                 !CLEAR THE ERROR FLG
5785 ;          3856        DAT_DM = ONE;                    !SET DATA DIAG MODE
5786 ;          3857        ML_FUNC = write;                 !LOAD MLCS1 WITH WRITE FUNCTION
5787 ;          3858        DAT_CLK = ONE;                   !DO A DATA CLK
5788 ;          3859
5789 ;          3860        if .REG_INIT_FLG IS_SET          !SEE IF CALLER IS REG INIT TEST
5790 ;          3861                                         !SET ERROR FLAG IF NEQ
5791 ;          3862        then
5792 ;          3863            begin
5793 ;          3864            CLR_MBUS;                    !CLEAR MBUS TO GENERATE INIT DATA
5794 ;          3865            DAT_DM = ONE;
5795 ;          3866            end;
5796 ;          3867
5797 ;          3868        WT_DATA = WRT_MASK;              !SAVE THE DATA WRITTEN TO THE REGISTER
5798 ;          3869        RD_DATA = .MLD1;                 !READ AND SAVE THE REGISTER
5799 ;          3870
5800 ;          3871        if .RD_DATA neq .WT_DATA then .ERR_FLG = ONE;       !READ MLD1 FOR WRT_MASK
5801 ;          3872
5802 ;          3873                                         !SET ERR_FLG IF NEQ
5803 ;          3874        CLR_MBUS;                        !CLEAR THE MBUS
5804 ;          3875        end;
5808
5809                                      .SBTTL  RD.D1
5813 021662  004167  162156          RD.D1:  JSR     R1,$SAVE2                       ;                       3802
5814 021666  005076  000010                  CLR     @10(SP)                         ; ERR.FLG               3855
5815 021672  152777  000010  171374          BISB    #10,@ML.REG+120                 ;                       3856
5816 021700  142777  000077  171246          BICB    #77,@ML.REG                     ;                       3857
5817 021706  152777  000061  171240          BISB    #61,@ML.REG
5818 021714  152777  000020  171352          BISB    #20,@ML.REG+120                 ;                       3858
5819 021722  026727  171210  000001          CMP     REG.INIT.FLG,#1                 ;                       3860
5820 021730  001017                          BNE     1$
5821 021732  152777  000040  171254          BISB    #40,@ML.REG+40                  ;                       3863
5822 021740  016701  171636                  MOV     ML.DUT,R1
5823 021744  042701  177770                  BIC     #177770,R1
5824 021750  142777  000007  171236          BICB    #7,@ML.REG+40
5825 021756  150177  171232                  BISB    R1,@ML.REG+40
5826 021762  152777  000010  171304          BISB    #10,@ML.REG+120                 ;                       3865
5827 021770  016600  000012          1$:     MOV     12(SP),R0                       ; INDEX,*               3868
5828 021774  006300                          ASL     R0
5829 021776  006300                          ASL     R0
5830 022000  006300                          ASL     R0
5831 022002  010001                          MOV     R0,R1
5832 022004  016100  013156                  MOV     ML.REG+2(R1),R0
5833 022010  056600  000014                  BIS     14(SP),R0                       ; TST.PAT,*
5834 022014  046100  013160                  BIC     ML.REG+4(R1),R0
```

```
5836                                    ;ML4                                                    14-May-1981 13:48:54    TOPS
5837                                    ;                                                       14-May-1981 13:05:08    PA:<
5838
5839 022020  016102  013162                      MOV     ML.REG+6(R1),R2
5840 022024  050002                              BIS     R0,R2
5841 022026  010267  171074                      MOV     R2,WT.DATA
5842 022032  017767  171306  171070              MOV     @ML.REG+170,RD.DATA             ;                       3869
5843 022040  026767  171064  171060              CMP     RD.DATA,WT.DATA                 ;                       3871
5844 022046  001403                              BEQ     2$
5845 022050  012776  000001  000010              MOV     #1,@10(SP)                      ; *,ERR.FLG
5846 022056  152777  000040  171130  2$:         BISB    #40,@ML.REG+40
5847 022064  016702  171512                      MOV     ML.DUT,R2
5848 022070  042702  177770                      BIC     #177770,R2
5849 022074  142777  000007  171112              BICB    #7,@ML.REG+40
5850 022102  150277  171106                      BISB    R2,@ML.REG+40
5851 022106  000207                              RTS     PC                              ;                       3802
5852
5853                                    ; Routine Size:  75 words
5854                                    ; Maximum stack depth per invocation:  3 words
5859
5860
```

```
 5862 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
 5863 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (41)
 5864
 5865 ;          3876  routine WRT_D2 (TST_PAT, index) : novalue =
 5866 ;          3877      begin
 5867 ;          3878
 5868 ;          3879  !++
 5869 ;          3880  ! FUNCTIONAL DESCRIPTION:
 5870 ;          3881  !       LOADS THE DATA DIAG REG 2 WITH A DATA PATTERN GENERATED BY THE MACRO WRT_MASK
 5871 ;          3882  ! FORMAL PARAMETERS:
 5872 ;          3883  !       TST_PAT
 5873 ;          3884  !       CURRENT DATA PATTERN TO BE LOADED IN THE REGISTER.
 5874 ;          3885  !       INDEX
 5875 ;          3886  !       USED BY THE MACRO WRT_MASK TO SELECT THE CURRENT REGISTERS ADDRESS,
 5876 ;          3887  !       FORCED HI, FORCED LO AND DON'T CARE MASK INFORMATION.
 5877 ;          3888  !--
 5878 ;          3889
 5879 ;          3890      DAT_DM = ONE;                          !SET DATA DIAG MODE
 5880 ;          3891      MLD2 = WRT_MASK;                       !LOAD MLD2 WITH GENERATED WRT_MASK PATTERN
 5881 ;          3892      DAT_DM = ZERO;                         !CLEAR DATA DIAG MODE
 5882 ;          3893      end;
 5886
 5887                                        .SBTTL  WRT.D2
 5891 022110  004167  161730          WRT.D2: JSR     R1,$SAVE2                       ;                       3876
 5892 022114  152777  000010  171152          BISB    #10,@ML.REG+120                 ;                       3890
 5893 022122  016600  000010                  MOV     10(SP),R0                       ; INDEX,*               3891
 5894 022126  006300                          ASL     R0
 5895 022130  006300                          ASL     R0
 5896 022132  006300                          ASL     R0
 5897 022134  010001                          MOV     R0,R1
 5898 022136  016100  013156                  MOV     ML.REG+2(R1),R0
 5899 022142  056600  000012                  BIS     12(SP),R0                       ; TST.PAT,*
 5900 022146  046100  013160                  BIC     ML.REG+4(R1),R0
 5901 022152  016102  013162                  MOV     ML.REG+6(R1),R2
 5902 022156  050002                          BIS     R0,R2
 5903 022160  010277  171170                  MOV     R2,@ML.REG+200
 5904 022164  142777  000010  171102          BICB    #10,@ML.REG+120                 ;                       3892
 5905 022172  000207                          RTS     PC                              ;                       3876
 5906
 5907                                ; Routine Size:  26 words
 5908                                ; Maximum stack depth per invocation:  3 words
 5913
 5914
```

```
5916 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
5917 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (42)
5918
5919 ;          3894  routine RD_D2 (TST_PAT, index, ERR_FLG) : novalue =
5920 ;          3895      begin
5921 ;          3896
5922 ;          3897  !++
5923 ;          3898  ! FUNCTIONAL DESCRIPTION:
5924 ;          3899  !
5925 ;          3900  !       COMPARE THE CONTENTS OF THE
5926 ;          3901  !       DATA DIAG REGISTER 2
5927 ;          3902  !       WITH THE MASKED DATA PATTERN
5928 ;          3903  !       GENERATED BY THE MACRO "WRT_MASK".
5929 ;          3904  !       IF THE COMPARE IS NOT EQUAL THEN THE
5930 ;          3905  !       FORMAL PARAMETER "ERR_FLG" IS
5931 ;          3906  !       ASSIGNED A ONE TO INDICATE THE
5932 ;          3907  !       ERROR.
5933 ;          3908  !
5934 ;          3909  ! FORMAL PARAMETERS:
5935 ;          3910  !
5936 ;          3911  !       TST_PAT
5937 ;          3912  !       DATA PATTERN TO BE MASKED AND
5938 ;          3913  !       COMPARED AGAINST THE CONTENTS
5939 ;          3914  !       OF THE REGISTER UNDER TEST.
5940 ;          3915  !
5941 ;          3916  !       INDEX
5942 ;          3917  !       USED BY THE MACRO WRT_MASK TO
5943 ;          3918  !       SELECT THE CURRENT REGISTERS ADDRESS,
5944 ;          3919  !       FORCED HI, FORCED LO AND DON'T CARE
5945 ;          3920  !       MASK INFOMATION.
5946 ;          3921  !
5947 ;          3922  !       ERR_FLG
5948 ;          3923  !       CONTAINS THE ADDRESS (PASSED BY REF)
5949 ;          3924  !       OF THE CALLERS ERROR FLG TO ENABLE THE
5950 ;          3925  !       CALLER TO EXAMINE THE ERROR STATUS
5951 ;          3926  !       OF THE ROUTINE CALL.
5952 ;          3927  !
5953 ;          3928  ! IMPLICIT INPUTS:
5954 ;          3929  !       WT_DATA
5955 ;          3930  !         GETS LOADED WITH THE GENERATED
5956 ;          3931  !         WRT_MASK DATA PATTERN THUS ALLOWING
5957 ;          3932  !         CALLER TO PRINT FAILING GOOD DATA.
5958 ;          3933  !
5959 ;          3934  !       RD_DATA
5960 ;          3935  !         GETS LOADED WITH DATA READ FROM THE
5961 ;          3936  !         REGISTER THUS ALLOWING CALLER
5962 ;          3937  !         TO PRINT FAILING BAD DATA.
5963 ;          3938  !
5964 ;          3939  ! IMPLICIT OUTPUTS:
5965 ;          3940  !       GLOBAL LOCATION WR_DATA
5966 ;          3941  !       AND RD_DATA LOADED WITH GOOD
5967 ;          3942  !       AND BAD REGISTER DATA
5968 ;          3943  !
5969 ;          3944  !--
5970 ;          3945
```

```
5972 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
5973 ;                                                       14-May-1981 13:05:08    FA:<NEALE>ML4.BLI.7 (42)
5974
5975 ;          3946      .ERR_FLG = ZERO;                   !CLEAR ERROR FLAG
5976 ;          3947      DAT_DM = ONE;                      !SET DATA DIAG MODE
5977 ;          3948      ML_FUNC = write;                   !LOAD WRITE FUNC TO CS1
5978 ;          3949      DAT_CLK = ONE;                     !DO A DATA CLOCK
5979 ;          3950
5980 ;          3951      if .REG_INIT_FLG IS_SET            !SEE IF CALLER IS REG INIT TEST
5981 ;          3952      then                               !CLEAR MBUS TO GENERATE INIT DATA
5982 ;          3953          begin
5983 ;          3954          CLR_MBUS;
5984 ;          3955          DAT_DM = ONE;
5985 ;          3956          end;
5986 ;          3957
5987 ;          3958      WT_DATA = WRT_MASK;                !SAVE THE DATA WRITTEN TO THE REGISTER
5988 ;          3959      RD_DATA = .MLD2;                   !READ AND SAVE THE REGISTER
5989 ;          3960
5990 ;          3961      if .RD_DATA neq .WT_DATA then .ERR_FLG = ONE;       !READ MLD2 FR WRT_MASK PATTERN
5991 ;          3962
5992 ;          3963                                         !SET ERROR FLAG IF NEQ
5993 ;          3964      CLR_MBUS;                          !CLR MASS BUSS
5994 ;          3965      end;
5998
5999                                        .SBTTL  RD.D2
6003 022174  004167  161644          RD.D2: JSR     R1,$SAVE2                ;                          3894
6004 022200  005076  000010                 CLR     @10(SP)                  ; ERR.FLG                  3946
6005 022204  152777  000010  171062         BISB    #10,@ML.REG+120          ;                          3947
6006 022212  142777  000077  170734         BICB    #77,@ML.REG              ;                          3948
6007 022220  152777  000061  170726         BISB    #61,@ML.REG
6008 022226  152777  000020  171040         BISB    #20,@ML.REG+120          ;                          3949
6009 022234  026727  170676  000001         CMP     REG.INIT.FLG,#1          ;                          3951
6010 022242  001017                         BNE     1$
6011 022244  152777  000040  170742         BISB    #40,@ML.REG+40           ;                          3953
6012 022252  016701  171324                 MOV     ML.DUT,R1
6013 022256  042701  177770                 BIC     #177770,R1
6014 022262  142777  000007  170724         BICB    #7,@ML.REG+40
6015 022270  150177  170720                 BISB    R1,@ML.REG+40
6016 022274  152777  000010  170772         BISB    #10,@ML.REG+120          ;                          3955
6017 022302  016600  000012           1$:   MOV     12(SP),R0                ; INDEX,*                  3958
6018 022306  006300                         ASL     R0
6019 022310  006300                         ASL     R0
6020 022312  006300                         ASL     R0
6021 022314  010001                         MOV     R0,R1
6022 022316  016100  013156                 MOV     ML.REG+2(R1),R0
6023 022322  056600  000014                 BIS     14(SP),R0                ; TST.PAT,*
6024 022326  046100  013160                 BIC     ML.REG+4(R1),R0
6025 022332  016102  013162                 MOV     ML.REG+6(R1),R2
6026 022336  050002                         BIS     R0,R2
```

```
                                        ;ML4                                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                                       14-May-1981 13:05:08    PA:<
                                                MOV     R2,WT.DATA
022344  017767  171004  170556                MOV     @ML.REG+200,RD.DATA             ;                                       3959
022352  026767  170552  170546                CMP     RD.DATA,WT.DATA                 ;                                       3961
022360  001403                                BEQ     2$
022362  012776  000001  000010                MOV     #1,@10(SP)                      ; *,ERR.FLG
022370  152777  000040  170616  2$:           BISB    #40,@ML.REG+40
022376  016702  171200                        MOV     ML.DUT,R2
022402  042702  177770                        BIC     #177770,R2
022406  142777  000007  170600                BICB    #7,@ML.REG+40
022414  150277  170574                        BISB    R2,@ML.REG+40
022420  000207                                RTS     PC                              ;                                       3894

                                        ; Routine Size:  75 words
                                        ; Maximum stack depth per invocation:  3 words
```

```
6052 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
6053 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (43)
6054
6055 ;          3966  routine WRT_D3 (TST_PAT, index) : novalue =
6056 ;          3967      begin
6057 ;          3968
6058 ;          3969  !++
6059 ;          3970  ! FUNCTIONAL DESCRIPTION:
6060 ;          3971  !
6061 ;          3972  !       LOADS THE DATA DIAG
6062 ;          3973  !       REGISTER 3 WITH A DATA PATTERN
6063 ;          3974  !       GENERATED BY THE MACRO
6064 ;          3975  !       WRT_MASK.
6065 ;          3976  !
6066 ;          3977  ! FORMAL PARAMETERS:
6067 ;          3978  !
6068 ;          3979  !       TST_PAT
6069 ;          3980  !       CURRENT DATA PATTERN TO BE
6070 ;          3981  !       LOADED IN THE REGISTER.
6071 ;          3982  !
6072 ;          3983  !       INDEX
6073 ;          3984  !       USE BY THE MACRO WRT_MASK
6074 ;          3985  !       TO SELECT THE CURRENT REGISTERS
6075 ;          3986  !       ADDRESS, FORCED HI, FORCED
6076 ;          3987  !       LO AND DON'T CARE MASK
6077 ;          3988  !       INFORMATION.
6078 ;          3989  !
6079 ;          3990  !--
6080 ;          3991
6081 ;          3992      DAT_DM = ONE;                      !SET DATA DIAG MODE
6082 ;          3993      MLE2_MASK = %o'000377';            !MASK OUT ECC CRC WORD BITS
6083 ;          3994      MLE2 = WRT_MASK;                   !LOAD MLE2 WITH GENERATED WRT_MASK PATTERN
6084 ;          3995      MLE2_MASK = %o'100300';            !RESTORE MASK
6085 ;          3996      DAT_DM = ZERO;                     !CLEAR DATA DIAG MODE.
6086 ;          3997      end;
6090
6091                                        .SBTTL  WRT.D3
6095 022422  004167  161416          WRT.D3: JSR     R1,$SAVE2                  ;                    3966
6096 022426  152777  000010  170640          BISB    #10,@ML.REG+120            ;                    3992
6097 022434  012767  000377  170700          MOV     #377,ML.REG+166            ;                    3993
6098 022442  016600  000010                  MOV     10(SP),R0                  ; INDEX,*            3994
6099 022446  006300                          ASL     R0
6100 022450  006300                          ASL     R0
6101 022452  006300                          ASL     R0
6102 022454  010001                          MOV     R0,R1
6103 022456  016100  013156                  MOV     ML.REG+2(R1),R0
6104 022462  056600  000012                  BIS     12(SP),R0                  ; TST.PAT,*
6105 022466  046100  013160                  BIC     ML.REG+4(R1),R0
6106 022472  016102  013162                  MOV     ML.REG+6(R1),R2
```

```
6108                                    ;ML4                                                    14-May-1981 13:48:54    TOPS
6109                                    ;                                                       14-May-1981 13:05:08    PA:<
6110
6111 022476  050002                             BIS     R0,R2
6112 022500  010277  170630                      MOV     R2,@ML.REG+160
6113 022504  012767  100300  170630              MOV     #-77500,ML.REG+166              ;       3995
6114 022512  142777  000010  170554              BICB    #10,@ML.REG+120                 ;       3996
6115 022520  000207                              RTS     PC                              ;       3966
6116
6117                                    ; Routine Size:  32 words
6118                                    ; Maximum stack depth per invocation:  3 words
6123
6124
```

```
6126 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
6127 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (44)
6128
6129 ;          3998  routine RD_D3 (TST_PAT, index, ERR_FLG) : novalue =
6130 ;          3999      begin
6131 ;          4000
6132 ;          4001  !++
6133 ;          4002  ! FUNCTIONAL DESCRIPTION:
6134 ;          4003  !
6135 ;          4004  !       COMPARES THE CONTENTS OF THE
6136 ;          4005  !       DATA DIAG REGISTER 3
6137 ;          4006  !       WITH THE MASKED DATA PATTERN
6138 ;          4007  !       GENERATED BY THE MACRO 'WRT_MASK".
6139 ;          4008  !
6140 ;          4009  !       IF THE COMPARE IS NOT EQUAL THE
6141 ;          4010  !       FORMAL PARAMETER 'ERR_FLG" IS
6142 ;          4011  !       ASSIGNED A ONE TO INDICATE THE
6143 ;          4012  !       ERROR.
6144 ;          4013  !
6145 ;          4014  ! FORMAL PARAMETERS:
6146 ;          4015  !
6147 ;          4016  !       TST_PAT
6148 ;          4017  !       DATA PATTERN TO BE MASKED AND
6149 ;          4018  !       COMPARED AGAINST THE CONTENTS
6150 ;          4019  !       OF THE REGISTER UNDER TEST.
6151 ;          4020  !
6152 ;          4021  !       INDEX
6153 ;          4022  !       USED BY THE MACRO WRT_MASK TO
6154 ;          4023  !       SELECT THE CURRENT REGISTERS ADDRESS,
6155 ;          4024  !       FORCED HI, FORCED LO AND DON'T CARE
6156 ;          4025  !       MASK INFORMATION
6157 ;          4026  !
6158 ;          4027  !       ERR_FLG
6159 ;          4028  !       CONTAINS THE ADDRESS (PASSED BY REF)
6160 ;          4029  !       OF THE CALLERS ERROR FLG TO ENABLE THE
6161 ;          4030  !       CALLER TO EXAMINE THE ERROR STATUS
6162 ;          4031  !       OF THE ROUTINE CALL.
6163 ;          4032  !
6164 ;          4033  ! IMPLICIT INPUTS:
6165 ;          4034  !       WT_DATA
6166 ;          4035  !         GETS LOADED WITH THE GENERATED
6167 ;          4036  !         WRT_MASK DATA PATTERN THUS ALLOWING
6168 ;          4037  !         CALLER TO PRINT FAILING GOOD DATA.
6169 ;          4038  !
6170 ;          4039  !       RD_DATA
6171 ;          4040  !         GETS LOADED WITH DATA READ FROM THE
6172 ;          4041  !         REGISTER THUS ALLOWING CALLER
6173 ;          4042  !         TO PRINT FAILING BAD DATA.
6174 ;          4043  !
6175 ;          4044  ! IMPLICIT OUTPUTS:
6176 ;          4045  !       GLOBAL LOCATION WR_DATA
6177 ;          4046  !       AND RD_DATA LOADED WITH GOOD
6178 ;          4047  !       AND BAD REGISTER DATA
6179 ;          4048  !
6180 ;          4049  !--
```

```
6182 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
6183 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (44)
6184
6185 ;          4050
6186 ;          4051        .ERR_FLG = ZERO;                 !CLEAR ERROR FLAG
6187 ;          4052        MLE2_MASK = %o'000377';          !SET DATA DIAG MODF
6188 ;          4053        DAT_DM = ONE;
6189 ;          4054        ML_FUNC = write;                 !LOAD WRITE FUNCTION TO MLCS1
6190 ;          4055        DAT_CLK = ONE;                   !DO A DATA CLOCK
6191 ;          4056
6192 ;          4057        if .REG_INIT_FLG IS_SET          !SEE IF CALLER IS  REG INIT TEST
6193 ;          4058        then                             !CLEAR  MBUS TO GENERATE INIT DATA
6194 ;          4059            begin
6195 ;          4060            CLR_MBUS;
6196 ;          4061            DAT_DM = ONE;
6197 ;          4062            end;
6198 ;          4063
6199 ;          4064        WT_DATA = WRT_MASK;              !SAVE THE DATA WRITTEN TO THE REGISTER
6200 ;          4065        RD_DATA = .MLE2 or .IGNORE;      !READ AND SAVE THE REGISTER
6201 ;          4066
6202 ;          4067        if .WT_DATA neq .RD_DATA then .ERR_FLG = ONE;       !READ THE REG FOR WRT_MASK
6203 ;          4068
6204 ;          4069        MLE2_MASK = %o'100300';
6205 ;          4070        CLR_MBUS;                        !CLEAR THE MASS BUS
6206 ;          4071        end;
6210
6211                                          .SBTTL  RD.D3
6215 022522  004167  161316          RD.D3:  JSR     R1,$SAVE2                ;                          3998
6216 022526  005076  000010                  CLR     @10(SP)                  ; ERR.FLG                  4051
6217 022532  012767  000377  170602          MOV     #377,ML.REG+166          ;                          4052
6218 022540  152777  000010  170526          BISB    #10,@ML.REG+120          ;                          4053
6219 022546  142777  000077  170400          BICB    #77,@ML.REG              ;                          4054
6220 022554  152777  000061  170372          BISB    #61,@ML.REG
6221 022562  152777  000020  170504          BISB    #20,@ML.REG+120          ;                          4055
6222 022570  026727  170342  000001          CMP     REG.INIT.FLG,#1          ;                          4057
6223 022576  001017                          BNE     1$
6224 022600  152777  000040  170406          BISB    #40,@ML.REG+40           ;                          4059
6225 022606  016701  170770                  MOV     ML.DUT,R1
6226 022612  042701  177770                  BIC     #177770,R1
6227 022616  142777  000007  170370          BICB    #7,@ML.REG+40
6228 022624  150177  170364                  BISB    R1,@ML.REG+40
6229 022630  152777  000010  170436          BISB    #10,@ML.REG+120          ;                          4061
6230 022636  016600  000012          1$:     MOV     12(SP),R0                ; INDEX.*                  4064
6231 022642  006300                          ASL     R0
6232 022644  006300                          ASL     R0
6233 022646  006300                          ASL     R0
6234 022650  010001                          MOV     R0,R1
6235 022652  016100  013156                  MOV     ML.REG+2(R1),R0
6236 022656  056600  000014                  BIS     14(SP),R0                ; TST.PAT,*
```

```
                                                                                14-May-1981 13:48:54    TOPS
                                                                                14-May-1981 13:05:08    PA:<
                        ;ML4
                        ;
 6241 022662 046100 013160      BIC     ML.REG+4(R1),R0
 6242 022666 016102 013162      MOV     ML.REG+6(R1),R2
 6243 022672 050002             BIS     R0,R2
 6244 022674 010267 170226      MOV     R2,WT.DATA
 6245 022700 017702 170430      MOV     @ML.REG+160,R2                  ;                       4065
 6246 022704 056102 013162      BIS     ML.REG+6(R1),R2
 6247 022710 010267 170214      MOV     R2,RD.DATA
 6248 022714 026767 170206 170206       CMP     WT.DATA,RD.DATA         ;                       4067
 6249 022722 001403             BEQ     2$
 6250 022724 012776 000001 000010       MOV     #1,@10(SP)              ; *,ERR.FLG
 6251 022732 012767 100300 170402 2$:   MOV     #-77500,ML.REG+166      ;                       4069
 6252 022740 152777 000040 170246       BISB    #40,@ML.REG+40
 6253 022746 016702 170630      MOV     ML.DUT,R2
 6254 022752 042702 177770      BIC     #177770,R2
 6255 022756 142777 000007 170230       BICB    #7,@ML.REG+40
 6256 022764 150277 170224      BISB    R2,@ML.REG+40
 6257 022770 000207             RTS     PC                              ;                       3998
 6258
 6259                   ; Routine Size:  84 words
 6260                   ; Maximum stack depth per invocation:  3 words
 6265
 6266
```

```
6268 :ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
6269 :                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (45)
6270
6271 ;          4072  routine WRT_DS (TST_PAT, index) : novalue =
6272 ;          4073      begin
6273 ;          4074
6274 ;          4075  !++
6275 ;          4076  ! FUNCTIONAL DESCRIPTION:
6276 ;          4077  !       DUMMY ROUTINE CALL TO ASSIST IN THE READ
6277 ;          4078  !       WRITE REGISTER ALGORITHEM
6278 ;          4079  !
6279 ;          4080  ! FORMAL PARAMETERS:
6280 ;          4081  !       TST_PAT
6281 ;          4082  !       DATA PATTERN TO BE MASKED AND
6282 ;          4083  !       COMPARED AGAINST THE CONTENTS
6283 ;          4084  !       OF THE REGISTER UNDER TEST
6284 ;          4085  !
6285 ;          4086  !       INDEX
6286 ;          4087  !       USED BY THE MACRO WRT_MASK TO
6287 ;          4088  !       SELECT THE CURRENT REGISTERS ADDRESS,
6288 ;          4089  !       FORCED HI, FORCED LO AND DON'T CARE
6289 ;          4090  !       MASK INFORMATION.
6290 ;          4091  !
6291 ;          4092  !--
6292 ;          4093
6293 ;          4094  !
6294 ;          4095  !  DRIVE STATUS REG IS READ ONLY
6295 ;          4096      return;
6296 ;          4097      end;
6300
6301                                  .SBTTL  WRT.DS
6305 022772  000207                WRT.DS: RTS     PC                              ;                4072
6306
6307                                  ; Routine Size:  1 word
6308                                  ; Maximum stack depth per invocation:  0 words
6313
6314
```

```
6316 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
6317 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (46)
6318
6319 ;          4098  routine RD_DS (TST_PAT, index, ERR_FLG) : novalue =
6320 ;          4099      begin
6321 ;          4100
6322 ;          4101  !++
6323 ;          4102  ! FUNCTIONAL DESCRIPTION:
6324 ;          4103  !
6325 ;          4104  !       COMPARES THE CONTENTS OF THE
6326 ;          4105  !       DRIVE STATUS REGISTER WITH THE
6327 ;          4106  !       MASKED DATA PATTERN
6328 ;          4107  !       GENERATED BY THE MACRO 'WRT_MASK'.
6329 ;          4108  !
6330 ;          4109  !       IF THE COMPARE IS NOT EQUAL THEN
6331 ;          4110  !       THE FORMAL PARAMETER 'ERR_FLG' IS
6332 ;          4111  !       ASSIGNED A ONE TO INDICATE THE
6333 ;          4112  !       ERROR.
6334 ;          4113  !
6335 ;          4114  ! FORMAL PARAMETERS:
6336 ;          4115  !
6337 ;          4116  !       TST_PAT
6338 ;          4117  !       DATA PATTERN TO BE MASKED AND
6339 ;          4118  !       COMPARED AGAINST THE CONTENTS
6340 ;          4119  !       OF THE REGISTER UNDER TEST.
6341 ;          4120  !
6342 ;          4121  !       INDEX
6343 ;          4122  !       USED BY THE MACRO WRT_MASK TO
6344 ;          4123  !       SELECT THE CURRENT REGISTERS ADDRESS,
6345 ;          4124  !       FORCED HI, FORCED LO AND DON'T CARE
6346 ;          4125  !       MASK INFORMATION.
6347 ;          4126  !
6348 ;          4127  !       ERR_FLG
6349 ;          4128  !       CONTAINS THE ADDRESS (PASSED BY REF)
6350 ;          4129  !       OF THE CALLERS ERROR FLG TO ENABLE THE
6351 ;          4130  !       CALLER TO EXAMINE THE ERROR STATUS
6352 ;          4131  !       OF THE ROUTINE CALL.
6353 ;          4132  !
6354 ;          4133  ! IMPLICIT INPUTS:
6355 ;          4134  !       WT_DATA
6356 ;          4135  !         GETS LOADED WITH THE GENERATED
6357 ;          4136  !         WRT_MASK DATA PATTERN THUS ALLOWING
6358 ;          4137  !         CALLER TO PRINT FAILING GOOD DATA.
6359 ;          4138  !
6360 ;          4139  !       RD_DATA
6361 ;          4140  !         GETS LOADED WITH DATA READ FROM THE
6362 ;          4141  !         REGISTER THUS ALLOWING CALLER
6363 ;          4142  !         TO PRINT FAILING BAD DATA.
6364 ;          4143  !
6365 ;          4144  ! IMPLICIT OUTPUTS:
6366 ;          4145  !       GLOBAL LOCATION WR_DATA
6367 ;          4146  !       AND RD_DATA LOADED WITH GOOD
6368 ;          4147  !       AND BAD REGISTER DATA
6369 ;          4148  !
6370 ;          4149  !--
```

```
6372 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
6373 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (46)
6374
6375 ;          4150
6376 ;          4151      .ERR_FLG = ZERO;                  !CLEAR THE ERROR FLAG
6377 ;          4152      WT_DATA = WRT_MASK;               !SAVE THE DATA WRITTEN TO THE REGISTER
6378 ;          4153      RD_DATA = .MLDS or .IGNORE;       !READ AND SAVE THE REGISTER
6379 ;          4154
6380 ;          4155      if .WT_DATA neq .RD_DATA then .ERR_FLG = ONE;       !READ THE REG FOR WRT_MASK
6381 ;          4156
6382 ;          4157                                        !SET ERROR FLAG IF NEQ
6383 ;          4158      end;
6387
6388                                    .SBTTL  RD.DS
6392 022774  004167  161044          RD.DS:  JSR     R1,$SAVE2                   ;                          4098
6393 023000  005076  000010                  CLR     @10(SP)                     ; ERR.FLG                  4151
6394 023004  016600  000012                  MOV     12(SP),R0                   ; INDEX,*                  4152
6395 023010  006300                          ASL     R0
6396 023012  006300                          ASL     R0
6397 023014  006300                          ASL     R0
6398 023016  010001                          MOV     R0,R1
6399 023020  016100  013156                  MOV     ML.REG+2(R1),R0
6400 023024  056600  000014                  BIS     14(SP),R0                   ; TST.PAT,*
6401 023030  046100  013160                  BIC     ML.REG+4(R1),R0
6402 023034  016102  013162                  MOV     ML.REG+6(R1),R2
6403 023040  050002                          BIS     R0,R2
6404 023042  010267  170060                  MOV     R2,WT.DATA
6405 023046  017702  170152                  MOV     @ML.REG+50,R2               ;                          4153
6406 023052  056102  013162                  BIS     ML.REG+6(R1),R2
6407 023056  010267  170046                  MOV     R2,RD.DATA
6408 023062  026767  170040  170040          CMP     WT.DATA,RD.DATA             ;                          4155
6409 023070  001403                          BEQ     1$
6410 023072  012776  000001  000010          MOV     #1,@10(SP)                  ; *,ERR.FLG
6411 023100  000207                  1$:     RTS     PC                          ;                          4098
6412
6413                                    ; Routine Size:  35 words
6414                                    ; Maximum stack depth per invocation:  3 words
6419
6420
```

```
6422 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
6423 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (47)
6424
6425 ;          4159  routine WRT_REG (TST_PAT, REG_SEL, index) : novalue =
6426 ;          4160      begin
6427 ;          4161
6428 ;          4162  !++
6429 ;          4163  ! FUNCTIONAL DESCRIPTION:
6430 ;          4164  !       A ROUTINE TO SELECTIVELY
6431 ;          4165  !       CALLED ROUTINES WHICH
6432 ;          4166  !       WRITE TO INDIVIDUAL ML11
6433 ;          4167  !       REGISTERS
6434 ;          4168  !
6435 ;          4169  ! FORMAL PARAMETERS:
6436 ;          4170  !       REG_SEL
6437 ;          4171  !       CASE SELECT EXPRESSION TO
6438 ;          4172  !       SELECT THE WRITE REGISTER
6439 ;          4173  !       ROUTINE TO CALLED
6440 ;          4174  !
6441 ;          4175  !       TST_PAT
6442 ;          4176  !       DATA PATTERN WHICH THE SELECTED
6443 ;          4177  !       REGISTER WILL BE TESTED AGAINST
6444 ;          4178  !
6445 ;          4179  !       INDEX
6446 ;          4180  !       LOADED WITH THE ML_REG INDEX
6447 ;          4181  !       SELECT NUMBER OF THE REGISTER
6448 ;          4182  !       BEING TESTED
6449 ;          4183  !
6450 ;          4184  ! SIDE EFFECTS:
6451 ;          4185  !       WHEN A WRITE REGISTER ROUTINE IS CALLED
6452 ;          4186  !       THE VARIABLE 'INDEX' FROM THE CALLING
6453 ;          4187  !       TEST IS LOADED WITH THE REGISTERS
6454 ;          4188  !       ML_REG INDEX NUMBER.
6455 ;          4189  !
6456 ;          4190  !       THIS ENABLES THE CALLING TEST TO FIND
6457 ;          4191  !       THE FAILING REGISTER ADDRESS.
6458 ;          4192  !
6459 ;          4193  !--
6460 ;          4194
6461 ;          4195      case .REG_SEL from 0 to 13 of              !SELECT THE WRITE REGISTER ROUTINE CALL
6462 ;          4196          set
6463 ;          4197
6464 ;          4198          [0] :
6465 ;          4199              WRT_CS1 (.TST_PAT, .index = 0);    !CALL ROUTINE TO LOAD MLCS1
6466 ;          4200
6467 ;          4201          [1] :
6468 ;          4202              WRT_ER (.TST_PAT, .index = 6);     !CALL ROUTINE TO LOAD MLER
6469 ;          4203
6470 ;          4204          [2] :
6471 ;          4205              WRT_DA (.TST_PAT, .index = 3);     .CALL ROUTINE TO LOAD MLDA
6472 ;          4206
6473 ;          4207          [3] :
6474 ;          4208              WRT_MR (.TST_PAT, .index = 10);    !CALL ROUTINE TO LOAD MLMR
6475 ;          4209
6476 ;          4210          [4] :
```

```
                                                        14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
                                                        14-May-1981 13:05:08    PA:<NEALE>ML4.SLI.7 (47)
:ML4
:
;          4211          WRT_E1 (.TST_PAT, .index = 13);     !CALL ROUTINE TO LOAD MLE1
;          4212
;          4213      [5] :
;          4214          WRT_E2 (.TST_PAT, .index = 14);     !CALL ROUTINE TO LOAD MLE2
;          4215
;          4216      [6] :
;          4217          WRT_PA (.TST_PAT, .index = 8);      !CALL ROUTINE TO LOAD MLPA
;          4218
;          4219      [7] :
;          4220          WRT_PD (.TST_PAT, .index = 19);     !CALL ROUTINE TO LOAD MLPD
;          4221
;          4222      [8] :
;          4223          WRT_EE (.TST_PAT, .index = 17);     !CALL ROUTINE TO LOAD MLEE
;          4224
;          4225      [9] :
;          4226          WRT_EL (.TST_PAT, .index = 18);     !CALL ROUTINE TO LOAD MLEL
;          4227
;          4228     [10] :
;          4229          WRT_DS (.TST_PAT, .index = 5);      !CALL ROUTINE TO LOAD MLDS
;          4230
;          4231     [11] :
;          4232          WRT_D1 (.TST_PAT, .index = 15);     !CALL ROUTINE TO LOAD MLD1
;          4233
;          4234     [12] :
;          4235          WRT_D2 (.TST_PAT, .index = 16);     !CALL ROUTINE TO LOAD MLD2
;          4236
;          4237      [13] :
;          4238          WRT_D3 (.TST_PAT, .index = 14);     .CALL ROUTINE TO LOAD MLE2
;          4239      tes;
;          4240
;          4241  end;
6515
6516                                        .SBTTL  WRT.REG
6520 023102  004167  160736          WRT.REG:JSR     R1,$SAVE2                   ;                       4159
6521 023106  016600  000010                  MOV     10(SP),R0                   ; INDEX,*               4199
6522 023112  016601  000014                  MOV     14(SP),R1                   ; TST.PAT,*
6523 023116  016602  000012                  MOV     12(SP),R2                   ; REG.SEL,*             4195
6524 023122  006302                          ASL     R2
6525 023124  066207  023130                  ADD     1$(R2),PC
6526 023130  000034                  1$:     .WORD   2$-1$
6527 023132  000050                          .WORD   3$-1$
6528 023134  000066                          .WORD   4$-1$
6529 023136  000104                          .WORD   5$-1$
6530 023140  000122                          .WORD   6$-1$
6531 023142  000140                          .WORD   7$-1$
6532 023144  000156                          .WORD   8$-1$
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                                .WORD   9$-1$
                                                .WORD   10$-1$
                                                .WORD   11$-1$
                                                .WORD   12$-1$
                                                .WORD   13$-1$
                                                .WORD   14$-1$
                                                .WORD   15$-1$
023164  010146                  2$:     MOV     R1,-(SP)                ;                       4199
023166  005010                          CLR     (R0)
023170  005046                          CLR     -(SP)
023172  004767  174272                  JSR     PC,WRT.CS1
023176  000532                          BR      16$                     ;                       4195
023200  010146                  3$:     MOV     R1,-(SP)                ;                       4202
023202  012710  000006                  MOV     #6,(R0)
023206  011046                          MOV     (R0),-(SP)
023210  004767  174432                  JSR     PC,WRT.ER
023214  000523                          BR      16$                     ;                       4195
023216  010146                  4$:     MOV     R1,-(SP)                ;                       4205
023220  012710  000003                  MOV     #3,(R0)
023224  011046                          MOV     (R0),-(SP)
023226  004767  174572                  JSR     PC,WRT.DA
023232  000514                          BR      16$                     ;                       4195
023234  010146                  5$:     MOV     R1,-(SP)                ;                       4208
023236  012710  000012                  MOV     #12,(R0)
023242  011046                          MOV     (R0),-(SP)
023244  004767  174732                  JSR     PC,WRT.MR
023250  000505                          BR      16$                     ;                       4195
023252  010146                  6$:     MOV     R1,-(SP)                ;                       4211
023254  012710  000015                  MOV     #15,(R0)
023260  011046                          MOV     (R0),-(SP)
023262  004767  175300                  JSR     PC,WRT.E1
023266  000476                          BR      16$                     ;                       4195
023270  010146                  7$:     MOV     R1,-(SP)                ;                       4214
023272  012710  000016                  MOV     #16,(R0)
023276  011046                          MOV     (R0),-(SP)
023300  004767  175470                  JSR     PC,WRT.E2
023304  000467                          BR      16$                     ;                       4195
023306  010146                  8$:     MOV     R1,-(SP)                ;                       4217
023310  012710  000010                  MOV     #10,(R0)
023314  011046                          MOV     (R0),-(SP)
023316  004767  175036                  JSR     PC,WRT.PA
023322  000460                          BR      16$                     ;                       4195
023324  010146                  9$:     MOV     R1,-(SP)                ;                       4220
023326  012710  000023                  MOV     #23,(R0)
023332  011046                          MOV     (R0),-(SP)
023334  004767  175666                  JSR     PC,WRT.PD
023340  000451                          BR      16$                     ;                       4195
023342  010146                  10$:    MOV     R1,-(SP)                ;                       4223
023344  012710  000021                  MOV     #21,(R0)
023350  011046                          MOV     (R0),-(SP)
023352  004767  176156                  JSR     PC,WRT.EE
023356  000442                          BR      16$                     ;                       4195
```

```
6590                                    ;ML4                                                    14-May-1981 13:48:54    TOPS
6591                                    ;                                                       14-May-1981 13:05:08    PA:<
6592
6593 023360  010146                     11$:    MOV     R1,-(SP)                ;                                       4226
6594 023362  012710  000022                     MOV     #22,(R0)
6595 023366  011046                             MOV     (R0),-(SP)
6596 023370  004767  176076                     JSR     PC,WRT.EL
6597 023374  000433                             BR      16$                     ;                                       4195
6598 023376  010146                     12$:    MOV     R1,-(SP)                ;                                       4229
6599 023400  012710  000005                     MOV     #5,(R0)
6600 023404  011046                             MOV     (R0),-(SP)
6601 023406  004767  177360                     JSR     PC,WRT.DS
6602 023412  000424                             BR      16$                     ;                                       4195
6603 023414  010146                     13$:    MOV     R1,-(SP)                ;                                       4232
6604 023416  012710  000017                     MOV     #17,(R0)
6605 023422  011046                             MOV     (R0),-(SP)
6606 023424  004767  176146                     JSR     PC,WRT.D1
6607 023430  000415                             BR      16$                     ;                                       4195
6608 023432  010146                     14$:    MOV     R1,-(SP)                ;                                       4235
6609 023434  012710  000020                     MOV     #20,(R0)
6610 023440  011046                             MOV     (R0),-(SP)
6611 023442  004767  176442                     JSR     PC,WRT.D2
6612 023446  000406                             BR      16$                     ;                                       4195
6613 023450  010146                     15$:    MOV     R1,-(SP)                ;                                       4238
6614 023452  012710  000016                     MOV     #16,(R0)
6615 023456  011046                             MOV     (R0),-(SP)
6616 023460  004767  176736                     JSR     PC,WRT.D3
6617 023464  022626                     16$:    CMP     (SP)+,(SP)+             ;                                       4160
6618 023466  000207                             RTS     PC                      ;                                       4159
6619
6620                                    ; Routine Size:  123 words
6621                                    ; Maximum stack depth per invocation:  5 words
6626
6627
```

```
;ML4

routine RD_REG (TST_PAT, REG_SEL, ERR_FLG) : novalue =
    begin

!++
! FUNCTIONAL DESCRIPTION:
!       A ROUTINE TO SELECTIVELY
!       CALLED ROUTINES WHICH
!       READ TO INDIVIDUAL ML11
!       REGISTERS.
!
! FORMAL PARAMETERS:
!       ERR_FLG
!       CONTAINS THE ADDRESS (PASSED BY REF)
!       OF THE CALLERS ERROR FLG TO ENABLE THE
!       CALLER TO EXAMINE THE ERROR STATUS
!       OF THE ROUTINE CALL.
!
!       REG_SEL
!       CASE SELECT EXPRESSION TO
!       SELECT THE WRITE REGISTER
!       ROUTINE TO CALLED
!
!       TST_PAT
!       DATA PATTERN WHICH THE SELECTED
!       REGISTER WILL BE TESTED AGAINST
!
!--

    case .REG_SEL from 0 to 13 of               !SELECT THE READ REGISTER ROUTINE CALL
        set

        [0] :
            RD_CS1 (.TST_PAT, 0, .ERR_FLG);     !CALL ROOUTINE TO READ MLCS1

        [1] :
            RD_ER (.TST_PAT, 6, .ERR_FLG);      !CALL ROUTINE TO READ MLER

        [2] :
            RD_DA (.TST_PAT, 3, .ERR_FLG);      !CALL ROUTINE TO READ MLDA

        [3] :
            RD_MR (.TST_PAT, 10, .ERR_FLG);     !CALL ROUTINE TO READ MLMR

        [4] :
            RD_E1 (.TST_PAT, 13, .ERR_FLG);     !CALL ROUTINE TO READ MLE1

        [5] :
            RD_E2 (.TST_PAT, 14, .ERR_FLG);     !CALL ROUTINE TO READ MLE2

        [6] :
            RD_PA (.TST_PAT, 8, .ERR_FLG);      !CALL ROUTINE TO READ MLPA
```

```
   6685 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
   6686 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (48)
   6687
   6688 ;          4294          [7] :
   6689 ;          4295              RD_PD (.TST_PAT, 19, .ERR_FLG);     !CALL ROUTINE TO READ MLPD
   6690 ;          4296
   6691 ;          4297          [8] :
   6692 ;          4298              RD_EE (.TST_PAT, 17, .ERR_FLG);     !CALL ROUTINE TO READ MLEE
   6693 ;          4299
   6694 ;          4300          [9] :
   6695 ;          4301              RD_EL (.TST_PAT, 18, .ERR_FLG);     !CALL ROUTINE TO READ MLEL
   6696 ;          4302
   6697 ;          4303          [10] :
   6698 ;          4304              RD_DS (.TST_PAT, 5, .ERR_FLG);      !CALL ROUTINE TO READ MLDS
   6699 ;          4305
   6700 ;          4306          [11] :
   6701 ;          4307              RD_D1 (.TST_PAT, 15, .ERR_FLG);     !CALL ROUTINE TO READ MLD1
   6702 ;          4308
   6703 ;          4309          [12] :
   6704 ;          4310              RD_D2 (.TST_PAT, 16, .ERR_FLG);     !CALL ROUTINE TO READ MLD2
   6705 ;          4311
   6706 ;          4312          [13] :
   6707 ;          4313              RD_D3 (.TST_PAT, 14, .ERR_FLG);     !CALL ROUTINE TO READ MLE2
   6708 ;          4314          tes;
   6709 ;          4315
   6710 ;          4316      end;
   6714
   6715                                  .SBTTL  RD.REG
   6719 023470  004167  160350      RD.REG: JSR     R1,$SAVE2                ;                                  4242
   6720 023474  016600  000010              MOV     10(SP),R0                ; ERR.FLG,*                        4274
   6721 023500  016601  000014              MOV     14(SP),R1                ; TST.PAT,*
   6722 023504  016602  000012              MOV     12(SP),R2                ; REG.SEL,*                        4270
   6723 023510  006302                      ASL     R2
   6724 023512  066207  023516              ADD     1$(R2),PC
   6725 023516  000034                1$:   .WORD   2$-1$
   6726 023520  000050                      .WORD   3$-1$
   6727 023522  000066                      .WORD   4$-1$
   6728 023524  000104                      .WORD   5$-1$
   6729 023526  000122                      .WORD   6$-1$
   6730 023530  000140                      .WORD   7$-1$
   6731 023532  000156                      .WORD   8$-1$
   6732 023534  000174                      .WORD   9$-1$
   6733 023536  000212                      .WORD   10$-1$
   6734 023540  000230                      .WORD   11$-1$
   6735 023542  000246                      .WORD   12$-1$
   6736 023544  000264                      .WORD   13$-1$
   6737 023546  000302                      .WORD   14$-1$
   6738 023550  000320                      .WORD   15$-1$
   6739 023552  010146                2$:   MOV     R1,-(SP)                 ;                                  4274
```

```
6741                                    ;ML4
6742                                    ;
6743
6744 023554  005046                            CLR     -(SP)
6745 023556  010046                            MOV     R0,-(SP)
6746 023560  004767  173754                    JSR     PC,RD.CS1
6747 023564  000532                            BR      16$
6748 023566  010146                    3$:     MOV     R1,-(SP)
6749 023570  012746  000006                    MOV     #6,-(SP)
6750 023574  010046                            MOV     R0,-(SP)
6751 023576  004767  174114                    JSR     PC,RD.ER
6752 023602  000523                            BR      16$
6753 023604  010146                    4$:     MOV     R1,-(SP)
6754 023606  012746  000003                    MOV     #3,-(SP)
6755 023612  010046                            MOV     R0,-(SP)
6756 023614  004767  174254                    JSR     PC,RD.DA
6757 023620  000514                            BR      16$
6758 023622  010146                    5$:     MOV     R1,-(SP)
6759 023624  012746  000012                    MOV     #12,-(SP)
6760 023630  010046                            MOV     R0,-(SP)
6761 023632  004767  174414                    JSR     PC,RD.MR
6762 023636  000505                            BR      16$
6763 023640  010146                    6$:     MOV     R1,-(SP)
6764 023642  012746  000015                    MOV     #15,-(SP)
6765 023646  010046                            MOV     R0,-(SP)
6766 023650  004767  174776                    JSR     PC,RD.E1
6767 023654  000476                            BR      16$
6768 023656  010146                    7$:     MOV     R1,-(SP)
6769 023660  012746  000016                    MOV     #16,-(SP)
6770 023664  010046                            MOV     R0,-(SP)
6771 023666  004767  175200                    JSR     PC,RD.E2
6772 023672  000467                            BR      16$
6773 023674  010146                    8$:     MOV     R1,-(SP)
6774 023676  012746  000010                    MOV     #10,-(SP)
6775 023702  010046                            MOV     R0,-(SP)
6776 023704  004767  174534                    JSR     PC,RD.PA
6777 023710  000460                            BR      16$
6778 023712  010146                    9$:     MOV     R1,-(SP)
6779 023714  012746  000023                    MOV     #23,-(SP)
6780 023720  010046                            MOV     R0,-(SP)
6781 023722  004767  175406                    JSR     PC,RD.PD
6782 023726  000451                            BR      16$
6783 023730  010146                    10$:    MOV     R1,-(SP)
6784 023732  012746  000021                    MOV     #21,-(SP)
6785 023736  010046                            MOV     R0,-(SP)
6786 023740  004767  175572                    JSR     PC,RD.EE
6787 023744  000442                            BR      16$
6788 023746  010146                    11$:    MOV     R1,-(SP)
6789 023750  012746  000022                    MOV     #22,-(SP)
6790 023754  010046                            MOV     R0,-(SP)
6791 023756  004767  175512                    JSR     PC,RD.EL
6792 023762  000433                            BR      16$
6793 023764  010146                    12$:    MOV     R1,-(SP)
6794 023766  012746  000005                    MOV     #5,-(SP)
6795 023772  010046                            MOV     R0,-(SP)
```

14-May-1981 13:48:54 TOPS
14-May-1981 13:05:08 PA:<

```
;                                                                       4270
;                                                                       4277
;                                                                       4270
;                                                                       4280
;                                                                       4270
;                                                                       4283
;                                                                       4270
;                                                                       4286
;                                                                       4270
;                                                                       4289
;                                                                       4270
;                                                                       4292
;                                                                       4270
;                                                                       4295
;                                                                       4270
;                                                                       4298
;                                                                       4270
;                                                                       4301
;                                                                       4270
;                                                                       4304
```

```
                                ;ML4                                                    14-May-1981 13:48:54    TOPS
                                ;                                                       14-May-1981 13:05:08    PA:<
 023774  004767  176774                JSR     PC,RD.DS
 024000  000424                        BR      16$                             ;                               4270
 024002  010146                13$:    MOV     R1,-(SP)                        ;                               4307
 024004  012746  000017                MOV     #17,-(SP)
 024010  010046                        MOV     R0,-(SP)
 024012  004767  175644                JSR     PC,RD.D1
 024016  000415                        BR      16$                             ;                               4270
 024020  010146                14$:    MOV     R1,-(SP)                        ;                               4310
 024022  012746  000020                MOV     #20,-(SP)
 024026  010046                        MOV     R0,-(SP)
 024030  004767  176140                JSR     PC,RD.D2
 024034  000406                        BR      16$                             ;                               4270
 024036  010146                15$:    MOV     R1,-(SP)                        ;                               4313
 024040  012746  000016                MOV     #16,-(SP)
 024044  010046                        MOV     R0,-(SP)
 024046  004767  176450                JSR     PC,RD.D3
 024052  062706  000006        16$:    ADD     #6,SP                           ;                               4243
 024056  000207                        RTS     PC                              ;                               4242

                                ; Routine Size:  124 words
                                ; Maximum stack depth per invocation:  6 words

;         4317  !<BLF/PAGE>
```

```
 6829 :ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
 6830 :                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (49)
 6831
 6832 :          4318  BGNMSG (DUMPER);
 6836
 6837                                        .SBTTL  DUMPER
 6841 024060  004767  000004          DUMPER::JSR     PC,M$DUMPER                 ;                          4318
 6842 024064  104423                         TRAP    23
 6843 024066  000207                         RTS     PC
 6844
 6845                                ; Routine Size:  4 words
 6846                                ; Maximum stack depth per invocation:  0 words
 6851
 6852
 6853 :          4319
 6854 :          4320  !++
 6855 :          4321  !       FUNCTIONAL DESCRIPTION
 6856 :          4322  !               UPON COMPLETION OF ERROR MESSAGES
 6857 :          4323  !               DUMP OUT ALL PERTINENT DRIVE AND
 6858 :          4324  !               RH REGISTERS
 6859 :          4325  !--
 6860 :          4326
 6861 :          4327  PRINTB (ONE_FMT, PHR_14);                   !PRINT REGISTER DUMP MESSAGE
 6862 :          4328  PRINTB (FMT_23);                            !PRINT DUMPER COLUMN HEADINGS
 6863 :          4329  PRINTB (FMT_24, REG_1, MLCS1, .MLCS1);      !PRINT OUT THE RH & ML11 REGISTER CONTENTS
 6864 :          4330  PRINTB (FMT_24, REG_18, MLWC, .MLWC);
 6865 :          4331  PRINTB (FMT_24, REG_19, MLBA, .MLBA);
 6866 :          4332  PRINTB (FMT_24, REG_6, MLDA, .MLDA);
 6867 :          4333  PRINTB (FMT_24, REG_17, MLCS2, .MLCS2);
 6868 :          4334  PRINTB (FMT_24, REG_2, MLDS, .MLDS);
 6869 :          4335  PRINTB (FMT_24, REG_3, MLER, .MLER);
 6870 :          4336  PRINTB (FMT_24, REG_5, MLAS, .MLAS);
 6871 :          4337  PRINTB (FMT_24, REG_4, MLMR, .MLMR);
 6872 :          4338  PRINTB (FMT_24, REG_7, MLDT, .MLDT);
 6873 :          4339  PRINTB (FMT_24, REG_9, MLSN, .MLSN);
 6874 :          4340  PRINTB (FMT_24, REG_14, MLEE, .MLEE);
 6875 :          4341  PRINTB (FMT_24, REG_15, MLEL, .MLEL);
 6876 :          4342
 6877 :          4343  if .LST_DUT_REG eql 21                      !SEE IF THIS IS A RH70
 6878 :          4344  then                                        !IF YES THEN PRINT RH70 REGISTERS
 6879 :          4345      begin
 6880 :          4346      PRINTB (FMT_24, REG_20, MLBAE, .MLBAE);
 6881 :          4347      PRINTB (FMT_24, REG_21, MLCS3, .MLCS3);
 6882 :          4348      end;
 6883 :          4349
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (49)

```
6885 ;ML4
6886 ;
6887
6888 ;          4350  ENDMSG:
6892
6893                                    .SBTTL  MSDUMPER
6897 024070                        MSDUMPER:
6898 024070  012746  007662                MOV     #PHR.14,-(SP)                   ;                       4327
6899 024074  012746  006016                MOV     #ONE.FMT,-(SP)
6900 024100  012746  000002                MOV     #2,-(SP)
6901 024104  010600                        MOV     SP,R0                           ; SP,*
6902 024106  104414                        TRAP    14
6903 024110  012716  005652                MOV     #FMT.23,(SP)                    ;                       4328
6904 024114  012746  000001                MOV     #1,-(SP)
6905 024120  010600                        MOV     SP,R0                           ; SP,*
6906 024122  104414                        TRAP    14
6907 024124  017716  167024                MOV     @ML.REG,(SP)                    ;                       4329
6908 024130  016746  167020                MOV     ML.REG,-(SP)
6909 024134  012746  010256                MOV     #REG.1,-(SP)
6910 024140  012746  005722                MOV     #FMT.24,-(SP)
6911 024144  012746  000004                MOV     #4,-(SP)
6912 024150  010600                        MOV     SP,R0                           ; SP,*
6913 024152  104414                        TRAP    14
6914 024154  017716  167004                MOV     @ML.REG+10,(SP)                 ;                       4330
6915 024160  016746  167000                MOV     ML.REG+10,-(SP)
6916 024164  012746  010430                MOV     #REG.18,-(SP)
6917 024170  012746  005722                MOV     #FMT.24,-(SP)
6918 024174  012746  000004                MOV     #4,-(SP)
6919 024200  010600                        MOV     SP,R0                           ; SP,*
6920 024202  104414                        TRAP    14
6921 024204  017716  166764                MOV     @ML.REG+20,(SP)                 ;                       4331
6922 024210  016746  166760                MOV     ML.REG+20,-(SP)
6923 024214  012746  010436                MOV     #REG.19,-(SP)
6924 024220  012746  005722                MOV     #FMT.24,-(SP)
6925 024224  012746  000004                MOV     #4,-(SP)
6926 024230  010600                        MOV     SP,R0                           ; SP,*
6927 024232  104414                        TRAP    14
6928 024234  017716  166744                MOV     @ML.REG+30,(SP)                 ;                       4332
6929 024240  016746  166740                MOV     ML.REG+30,-(SP)
6930 024244  012746  010316                MOV     #REG.6,-(SP)
6931 024250  012746  005722                MOV     #FMT.24,-(SP)
6932 024254  012746  000004                MOV     #4,-(SP)
6933 024260  010600                        MOV     SP,R0                           ; SP,*
6934 024262  104414                        TRAP    14
6935 024264  017716  166724                MOV     @ML.REG+40,(SP)                 ;                       4333
6936 024270  016746  166720                MOV     ML.REG+40,-(SP)
6937 024274  012746  010420                MOV     #REG.17,-(SP)
6938 024300  012746  005722                MOV     #FMT.24,-(SP)
6939 024304  012746  000004                MOV     #4,-(SP)
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                        MOV     SP,R0                   ; SP,*
                                        TRAP    14
                                        MOV     @ML.REG+50,(SP)         ;                                               4334
                                        MOV     ML.REG+50,-(SP)
                                        MOV     #REG.2,-(SP)
                                        MOV     #FMT.24,-(SP)
                                        MOV     #4,-(SP)
                                        MOV     SP,R0                   ; SP,*
                                        TRAP    14
                                        MOV     @ML.REG+60,(SP)         ;                                               4335
                                        MOV     ML.REG+60,-(SP)
                                        MOV     #REG.3,-(SP)
                                        MOV     #FMT.24,-(SP)
                                        MOV     #4,-(SP)
                                        MOV     SP,R0                   ; SP,*
                                        TRAP    14
                                        MOV     @ML.REG+70,(SP)         ;                                               4336
                                        MOV     ML.REG+70,-(SP)
                                        MOV     #REG.5,-(SP)
                                        MOV     #FMT.24,-(SP)
                                        MOV     #4,-(SP)
                                        MOV     SP,R0                   ; SP,*
                                        TRAP    14
                                        MOV     @ML.REG+120,(SP)        ;                                               4337
                                        MOV     ML.REG+120,-(SP)
                                        MOV     #REG.4,-(SP)
                                        MOV     #FMT.24,-(SP)
                                        MOV     #4,-(SP)
                                        MOV     SP,R0                   ; SP,*
                                        TRAP    14
                                        MOV     @ML.REG+130,(SP)        ;                                               4338
                                        MOV     ML.REG+130,-(SP)
                                        MOV     #REG.7,-(SP)
                                        MOV     #FMT.24,-(SP)
                                        MOV     #4,-(SP)
                                        MOV     SP,R0                   ; SP,*
                                        TRAP    14
                                        MOV     @ML.REG+140,(SP)        ;                                               4339
                                        MOV     ML.REG+140,-(SP)
                                        MOV     #REG.9,-(SP)
                                        MOV     #FMT.24,-(SP)
                                        MOV     #4,-(SP)
                                        MOV     SP,R0                   ; SP,*
                                        TRAP    14
                                        MOV     @ML.REG+210,(SP)        ;                                               4340
                                        MOV     ML.REG+210,-(SP)
                                        MOV     #REG.14,-(SP)
                                        MOV     #FMT.24,-(SP)
                                        MOV     #4,-(SP)
                                        MOV     SP,R0                   ; SP,*
                                        TRAP    14
                                        ADD     #146,SP
```

```
                                        ;ML4                                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                                       14-May-1981 13:05:08    PA:<
                                                MOV     @ML.REG+220,(SP)                ;                                               4341
                                                MOV     ML.REG+220,-(SP)
                                                MOV     #REG.15,-(SP)
                                                MOV     #FMT.24,-(SP)
                                                MOV     #4,-(SP)
                                                MOV     SP,R0                           ; SP,*
                                                TRAP    14
                                                CMP     LST.DUT.REG,#25                 ;                               4343
                                                BNE     1$
                                                MOV     @ML.REG+240,-(SP)               ;                               4346
                                                MOV     ML.REG+240,-(SP)
                                                MOV     #REG.20,-(SP)
                                                MOV     #FMT.24,-(SP)
                                                MOV     #4,-(SP)
                                                MOV     SP,R0                           ; SP,*
                                                TRAP    14
                                                MOV     @ML.REG+250,(SP)                ;                               4347
                                                MOV     ML.REG+250,-(SP)
                                                MOV     #REG.21,-(SP)
                                                MOV     #FMT.24,-(SP)
                                                MOV     #4,-(SP)
                                                MOV     SP,R0                           ; SP,*
                                                TRAP    14
                                                ADD     #22,SP                          ;                               4345
                                        1$:     ADD     #12,SP                          ;                               4318
                                                RTS     PC

                                        ; Routine Size:  205 words
                                        ; Maximum stack depth per invocation:  52 words


                                ;       4351    !<BLF/PAGE>
```

```
7037 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
7038 :                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (50)
7039
7040 ;          4352  :
7041 ;          4353  BGNINIT;
7042 ;          4354  !       INITIALIZATION CODE IS EXECUTED AT THE BEGINNING OF EACH
7043 ;          4355  !       PASS, WHEN POWER DOWN/POWER UP HAS OCCURRED, OR WHEN THE
7044 ;          4356  !       OPERATOR HAS ISSUED A START, RESTART OR CONTINUE COMMAND.
7045 ;          4357  !         DURING INITIALIZATION, THE 'GPHARD' MACRO IS USED TO GET
7046 ;          4358  !       P-TABLE INFORMATION FOR THE LOGICAL UNIT UNDER TEST. THE
7047 ;          4359  !       NUMBER OF UNITS  AVAILABLE FOR TESTING IS CONTAINED IN A
7048 ;          4360  !       HEADER LOCATION ('L$UNIT').
7049 ;          4361
7050 ;          4362  local
7051 ;          4363      OFFSET;
7052 ;          4364
7053 ;          4365  external
7054 ;          4366      L$UNIT;
7055 ;          4367
7056 ;          4368  if not READEF (EF_CONTINUE)                    !SKIP INIT CODE IF CONTINUE
7057 ;          4369  then
7058 ;          4370      begin                                      !START GPHARDS AT LUN 0 AND LOAD 'ML_REG'
7059 ;          4371
7060 ;          4372      if READEF (EF_START)                       !SEE IF THIS IS THE VERY FIRST PASS
7061 ;          4373      then
7062 ;          4374          begin                                  !THIS IS CATEGORY 1 CODE
7063 ;          4375          ML_LUN = -1;
7064 ;          4376
7065 ;          4377          do
7066 ;          4378              begin
7067 ;          4379              ML_LUN = .ML_LUN + 1;              !INCREMENT LOGICAL UNIT NUMBER
7068 ;          4380
7069 ;          4381              if .ML_LUN geq .L$UNIT then DOCLN; !START OVER IF ALL UNITS HAVE BEEN TESTED
7070 ;          4382
7071 ;          4383              end
7072 ;          4384          until (GPHARD (.ML_LUN, PTBL_PTR)) neq 0;       !REPEAT UNTIL A ZERO IS NOT RETURNED
7073 ;          4385
7074 ;          4386          RH_ADD = .((.PTBL_PTR) + 0);           !GET BASE RH ADDRESS FOR THIS UNIT
7075 ;          4387          RH_TYP = .((.PTBL_PTR) + 2);           !GET RH TYPE FOR THIS UNIT
7076 ;          4388          RH_VEC = .((.PTBL_PTR) + 4);           !GET RH VECTOR FOR THIS UNIT
7077 ;          4389          OFFSET = 0;                            !INIT OFF SET COUNT
7078 ;          4390
7079 ;          4391          incr COUNT from 0 to 21 do             !LOAD THE REGISTER ADDRESS FOR THIS UNIT INTO ML_REG
7080 ;          4392              begin
7081 ;          4393              ML_REG [.COUNT, REGISTER_ADD] = .RH_ADD + .OFFSET;
7082 ;          4394              OFFSET = .OFFSET + 2;
7083 ;          4395              end;
7084 ;          4396
7085 ;          4397          end
7086 ;          4398      else                                       !IS THIS A NEW PASS
7087 ;          4399          begin
7088 ;          4400
7089 ;          4401          if READEF (EF_NEW) then ML_LUN = -1;   !IF NEW PASS START GPHARDS AT LUN 0
7090 ;          4402
7091 ;          4403          do
7092 ;ML4                                                        14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
7093 :                                                          14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (50)
```

```
7094
7095 ;          4404              begin
7096 ;          4405              ML_LUN = .ML_LUN + 1;                   !IF NOT GET NEXT LUN PTABLE
7097 ;          4406
7098 ;          4407              if .ML_LUN geq .L$UNIT then DOCLN;      !START OVER IF ALL UNITS ARE TESTED
7099 ;          4408
7100 ;          4409              end
7101 ;          4410          until (GPHARD (.ML_LUN, PTBL_PTR)) neq 0;       !REPEAT UNTIL A ZERO IS NOT RETURNED
7102 ;          4411
7103 ;          4412          end;
7104 ;          4413
7105 ;          4414      PAR_DIS = .((.PTBL_PTR) + 12);                  !GET PARITY DISABLE FLAG
7106 ;          4415      ML_DUT = .((.PTBL_PTR) + 10);                   !GET DRIVE NUMBER
7107 ;          4416      OP_NUM_ARR = .((.PTBL_PTR) + 6) - 1;            !GET OPERATOR NUMBER OF ARRAYS
7108 ;          4417      GOOD_BLK = ZEROES;                              !INIT GOOD BLOCK TO BLOCK ZERO
7109 ;          4418      ARR_16 = ZEROES;                                !INIT ARRAY 16 TO ZERO
7110 ;          4419      LST_ARR = ZEROES;                               !INIT LAST ARRAY TO ZERO
7111 ;          4420      LST_BLK = ZEROES;                               !INIT LAST BLOCK TO ZERO
7112 ;          4421
7113 ;          4422      if .((.PTBL_PTR) + 8) IS_SET                    !CALCULATE ML11 16K MOS RAM PARAMETERS
7114 ;          4423      then
7115 ;          4424          begin
7116 ;          4425          DRIVE_TYPE = %o'000110';                    !EXPECTED DRIVE TYPE VALUE
7117 ;          4426          W_C_SIZE = %o'140000';                      !WORD COUNT SIZE FOR 16K WORD XFER
7118 ;          4427          RAS_INC = %o'200';                          !RAS INCREMENT FOR 16K RAMS
7119 ;          4428          CHIP_SIZ = 16;                              !CHIP SIZE
7120 ;          4429          ARR_INC = %o'1000';                         !ARRAY INCREMENT
7121 ;          4430          ARR_16<9, 4> = %o'17';                      !ARRAY 16
7122 ;          4431          LST_ARR<9, 4> = .OP_NUM_ARR;                !LAST ARRAY
7123 ;          4432          LST_BLK<9, 4> = .OP_NUM_ARR;                !LAST BLOCK
7124 ;          4433          LST_BLK = .LST_BLK or %o'777';
7125 ;          4434          end
7126 ;          4435      else                                            !CALCULATE ML11 64K MOS RAM PARAMETERS
7127 ;          4436          begin
7128 ;          4437          DRIVE_TYPE = %o'000111';                    !EXPECTED DRIVE TYPE VALUE
7129 ;          4438          W_C_SIZE = %o'000000';                      !WORD COUNT SIZE FOR 64K WORD XFER
7130 ;          4439          RAS_INC = %o'1000';                         !RAS INCREMENT FOR 64K RAMS
7131 ;          4440          CHIP_SIZ = 64;                              !CHIP SIZE
7132 ;          4441          ARR_INC = %o'4000';                         !ARRAY INCREMENT
7133 ;          4442          ARR_16<11, 4> = %o'74';                     !ARRAY 16
7134 ;          4443          LST_ARR<11, 4> = .OP_NUM_ARR;               !LAST ARRAY
7135 ;          4444          LST_BLK<11, 4> = .OP_NUM_ARR;               !LAST BLOCK
7136 ;          4445          LST_BLK = .LST_BLK or %o'3777';
7137 ;          4446          end;
7138 ;          4447
7139 ;          4448      if .((.PTBL_PTR) + 2) eql %o'70' then LST_DUT_REG = 21 else LST_DUT_REG = 19;
7140 ;          4449
7141 ;          4450      PRINTB (FMT_17, .ML_LUN);                       !TELL OPERATOR WHICH UNIT IS BEING TESTED
7142 ;          4451      CLR_MBUS;                                       !CLEAR MASS BUS
7143 ;          4452      end;
7144 ;          4453
7145 ;          4454  ENDINIT;
7152
7153                                  .GLOBL  L$UNIT
7154
7155
7156                                  .SBTTL  LINIT
```

```
7160 024722  004167  157132          LINIT:  JSR     R1,$SAVE3                       ;                               4350
7161 024726  012700  000036                  MOV     #36,R0                          ;                               4368
7162 024732  104447                          TRAP    47
7163 024734  103001                          BHIS    1$
7164 024736  000207                          RTS     PC
7165 024740  012700  000040          1$:     MOV     #40,R0                          ;                               4372
7166 024744  104447                          TRAP    47
7167 024746  103061                          BHIS    5$
7168 024750  012767  177777  166622          MOV     #-1,ML.LUN                      ;                               4375
7169 024756  005267  166616          2$:     INC     ML.LUN                          ;                               4379
7170 024762  026767  166612  155022          CMP     ML.LUN,L$UNIT                   ;                               4381
7171 024770  002401                          BLT     3$
7172 024772  104444                          TRAP    44
7173 024774  016700  166600          3$:     MOV     ML.LUN,R0                       ;                               4384
7174 025000  104442                          TRAP    42
7175 025002  010067  164550                  MOV     R0,PTBL.PTR
7176 025006  005767  164544                  TST     PTBL.PTR
7177 025012  001761                          BEQ     2$
7178 025014  017767  164536  166550          MOV     @PTBL.PTR,RH.ADD                ;                               4386
7179 025022  016701  164530                  MOV     PTBL.PTR,R1                     ;                               4387
7180 025026  016167  000002  166540          MOV     2(R1),RH.TYP
7181 025034  016701  164516                  MOV     PTBL.PTR,R1                     ;                               4388
7182 025040  016167  000004  166530          MOV     4(R1),RH.VEC
7183 025046  005002                          CLR     R2                              ; OFFSET                        4389
7184 025050  005001                          CLR     R1                              ; COUNT                         4391
7185 025052  010100          4$:     MOV     R1,R0                           ; COUNT,*                       4393
7186 025054  006300                          ASL     R0
7187 025056  006300                          ASL     R0
7188 025060  006300                          ASL     R0
7189 025062  016703  166504                  MOV     RH.ADD,R3
7190 025066  060203                          ADD     R2,R3                           ; OFFSET,*
7191 025070  010360  013154                  MOV     R3,ML.REG(R0)
7192 025074  062702  000002                  ADD     #2,R2                           ; *,OFFSET                      4394
7193 025100  005201                          INC     R1                              ; COUNT                         4391
7194 025102  020127  000025                  CMP     R1,#25                          ; COUNT,*
7195 025106  003761                          BLE     4$
7196 025110  000426                          BR      8$                              ;                               4372
7197 025112  012700  000035          5$:     MOV     #35,R0                          ;                               4401
7198 025116  104447                          TRAP    47
7199 025120  103003                          BHIS    6$
7200 025122  012767  177777  166450          MOV     #-1,ML.LUN
7201 025130  005267  166444          6$:     INC     ML.LUN                          ;                               4405
7202                                 :ML4                                                    14-May-1981 13:48:54    TOPS
7203                                 :                                                       14-May-1981 13:05:08    PA:<
7204
7205 025134  026767  166440  154650          CMP     ML.LUN,L$UNIT                   ;                               4407
7206 025142  002401                          BLT     7$
7207 025144  104444                          TRAP    44
7208 025146  016700  166426          7$:     MOV     ML.LUN,R0                       ;                               4410
7209 025152  104442                          TRAP    42
7210 025154  010067  164376                  MOV     R0,PTBL.PTR
7211 025160  005767  164372                  TST     PTBL.PTR
7212 025164  001761                          BEQ     6$
7213 025166  016701  164364          8$:     MOV     PTBL.PTR,R1                     ;                               4414
7214 025172  016167  000014  164366          MOV     14(R1),PAR.DIS
7215 025200  016701  164352                  MOV     PTBL.PTR,R1                     ;                               4415
7216 025204  016167  000012  166370          MOV     12(R1),ML.DUT
```

```
7217 025212 016701 164340          MOV   PTBL.PTR,R1          ;                    4416
7218 025216 016103 000006          MOV   6(R1),R3
7219 025222 005303                 DEC   R3
7220 025224 010367 164330          MOV   R3,OP.NUM.ARR
7221 025230 005067 164330          CLR   GOOD.BLK             ;                    4417
7222 025234 005067 164334          CLR   ARR.16               ;                    4418
7223 025240 005067 164332          CLR   LST.ARR              ;                    4419
7224 025244 005067 164322          CLR   LST.BLK              ;                    4420
7225 025250 016701 164302          MOV   PTBL.PTR,R1          ;                    4422
7226 025254 026127 000010 000001   CMP   10(R1),#1
7227 025262 001054                 BNE   9$
7228 025264 012767 000110 165640   MOV   #110,DRIVE.TYPE      ;                    4425
7229 025272 012767 140000 165622   MOV   #-40000,W.C.SIZE     ;                    4426
7230 025300 012767 000200 165616   MOV   #200,RAS.INC         ;                    4427
7231 025306 012767 000020 164254   MOV   #20,CHIP.SIZ         ;                    4428
7232 025314 012767 001000 164240   MOV   #1000,ARR.INC        ;                    4429
7233 025322 052767 017000 164244   BIS   #17000,ARR.16        ;                    4430
7234 025330 016703 164224          MOV   OP.NUM.ARR,R3        ;                    4431
7235 025334 000303                 SWAB  R3
7236 025336 006303                 ASL   R3
7237 025340 042703 160777          BIC   #160777,R3
7238 025344 042767 017000 164224   BIC   #17000,LST.ARR
7239 025352 050367 164220          BIS   R3,LST.ARR
7240 025356 016703 164176          MOV   OP.NUM.ARR,R3        ;                    4432
7241 025362 000303                 SWAB  R3
7242 025364 006303                 ASL   R3
7243 025366 042703 160777          BIC   #160777,R3
7244 025372 042767 017000 164172   BIC   #17000,LST.BLK
7245 025400 050367 164166          BIS   R3,LST.BLK
7246 025404 052767 000777 164160   BIS   #777,LST.BLK         ;                    4433
7247 025412 000461                 BR    10$                  ;                    4422
7248 025414 012767 000111 165510 9$:  MOV  #111,DRIVE.TYPE    ;                    4437
7249 025422 005067 165474          CLR   W.C.SIZE             ;                    4438
7250 025426 012767 001000 165470   MOV   #1000,RAS.INC        ;                    4439
7251 025434 012767 000100 164126   MOV   #100,CHIP.SIZ        ;                    4440
7252 025442 012767 004000 164112   MOV   #4000,ARR.INC        ;                    4441
7253 025450 042767 074000 164116   BIC   #74000,ARR.16        ;                    4442
7254 025456 052767 060000 164110   BIS   #60000,ARR.16
7255 025464 016703 164070          MOV   OP.NUM.ARR,R3        ;                    4443
7256 025470 000303                 SWAB  R3
7257                          ;ML4                                  14-May-1981 13:48:54  TOPS
7258                          ;                                     14-May-1981 13:05:08  PA:<
7259
7260 025472 006303                 ASL   R3
7261 025474 006303                 ASL   R3
7262 025476 006303                 ASL   R3
7263 025500 042703 103777          BIC   #103777,R3
7264 025504 042767 074000 164064   BIC   #74000,LST.ARR
7265 025512 050367 164060          BIS   R3,LST.ARR
7266 025516 016703 164036          MOV   OP.NUM.ARR,R3        ;                    4444
7267 025522 000303                 SWAB  R3
7268 025524 006303                 ASL   R3
7269 025526 006303                 ASL   R3
7270 025530 006303                 ASL   R3
7271 025532 042703 103777          BIC   #103777,R3
7272 025536 042767 074000 164026   BIC   #74000,LST.BLK
7273 025544 050367 164022          BIS   R3,LST.BLK
```

```
 7274 025550  052767  003777  164014          BIS     #3777,LST.BLK                   ;                                       4445
 7275 025556  016701  163774          10$:    MOV     PTBL.PTR,R1                     ;                                       4448
 7276 025562  026127  000002  000070          CMP     2(R1),#70
 7277 025570  001004                          BNE     11$
 7278 025572  012767  000025  165334          MOV     #25,LST.DUT.REG
 7279 025600  000403                          BR      12$
 7280 025602  012767  000023  165324  11$:    MOV     #23,LST.DUT.REG
 7281 025610  016746  165764          12$:    MOV     ML.LUN,-(SP)                    ;                                       4450
 7282 025614  012746  005306                  MOV     #FMT.17,-(SP)
 7283 025620  012746  000002                  MOV     #2,-(SP)
 7284 025624  010600                          MOV     SP,R0                           ; SP,*
 7285 025626  104414                          TRAP    14
 7286 025630  152777  000040  165356          BISB    #40,@ML.REG+40
 7287 025636  016703  165740                  MOV     ML.DUT,R3
 7288 025642  042703  177770                  BIC     #177770,R3
 7289 025646  142777  000007  165340          BICB    #7,@ML.REG+40
 7290 025654  150377  165334                  BISB    R3,@ML.REG+40
 7291 025660  062706  000006                  ADD     #6,SP                           ;                                       4370
 7292 025664  000207                          RTS     PC                              ;                                       4350
 7293
 7294                                 ; Routine Size:  242 words
 7295                                 ; Maximum stack depth per invocation:  7 words
 7300
 7301
 7305
 7306                                         .SBTTL  L$INIT
 7310 025666  004767  177030          L$INIT::JSR     PC,LINIT                        ;                                       4452
 7311 025672  104411                          TRAP    11
 7312                                 ;ML4                                                    14-May-1981 13:48:54    TOPS
 7313                                 ;                                                       14-May-1981 13:05:08    PA:<
 7314
 7315 025674  000207                          RTS     PC
 7316
 7317                                 ; Routine Size:  4 words
 7318                                 ; Maximum stack depth per invocation:  0 words
 7323
 7324
 7325 ;          4455  !<BLF/PAGE>
```

```
7327 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
7328 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (51)
7329
7330 ;          4456  !
7331 ;          4457  BGNTST;
7332 ;          4458
7333 ;          4459  !++
7334 ;          4460  ! TEST NUMBER:  TST 1
7335 ;          4461  !
7336 ;          4462  ! TEST NAME:              MASS BUS READY TEST
7337 ;          4463  !
7338 ;          4464  ! TEST DESCRIPTION:
7339 ;          4465  !       TEST THE RH CONTROLLER FOR EXISTANCE
7340 ;          4466  !       BY:
7341 ;          4467  !
7342 ;          4468  !       1.  WRITTING ONES TO THE RH CS2 REGISTER
7343 ;          4469  !           RESULTING IN A MASS BUS CLEAR
7344 ;          4470  !
7345 ;          4471  !       2.  THEN READ THE CS2 REGISTER FOR CLEAR DATA
7346 ;          4472  !
7347 ;          4473  !--
7348 ;          4474
7349 ;          4475  CLR_MBUS;
7350 ;          4476  MLCS1 = ZEROES;                          !CLEAR OUT CS1
7351 ;          4477  MLCS2 = %o'177770';                      !LOAD CS2 WITH ONES  AND FORCE A MBUS CLEAR
7352 ;          4478  DELAY (ONE_US);
7353 ;          4479
7354 ;          4480  if ((.MLCS2) and (%o'177670')) neq ZERO  !SEE IF CS2 GOT CLEARED
7355 ;          4481  then
7356 ;          4482      begin                                !REPORT ERROR IF NOT CLEARED
7357 ;          4483      ERRDF (119, RH_ERROR, DUMPER);
7358 ;          4484      PRINTB (THR_FMT, REG_17, FNC_23, PHR_4);
7359 ;          4485      DODU (.ML_LUN);
7360 ;          4486      DOCLN;
7361 ;          4487      end;
7362 ;          4488
7363 ;          4489  ENDTST;
7367
7368                                          .SBTTL  $T1
7372 025676  010146                  $T1:    MOV     R1,-(SP)            ;                          4454
7373 025700  005746                          TST     -(SP)
7374 025702  152777  000040  165304          BISB    #40,@ML.REG+40      ;                          4457
7375 025710  016701  165666                  MOV     ML.DUT,R1
7376 025714  042701  177770                  BIC     #177770,R1
7377 025720  142777  000007  165266          BICB    #7,@ML.REG+40
7378 025726  150177  165262                  BISB    R1,@ML.REG+40
7379 025732  005077  165216                  CLR     @ML.REG             ;                          4476
7380 025736  012777  177770  165250          MOV     #-10,@ML.REG+40     ;                          4477
7381 025744  012700  000001                  MOV     #1,R0               ; *,$$TMP2                 4478
7382                              ;ML4                                                 14-May-1981 13:48:54    TOPS
7383                              ;                                                    14-May-1981 13:05:08    PA:<
7384
7385 025750  001410                  1$:     BEQ     4$
7386 025752  016701  154140                  MOV     L$DLY,R1            ; *,$$TMP1
7387 025756  001403                          BEQ     3$
7388 025760  005016                  2$:     CLR     (SP)                ; $$TMP
7389 025762  005301                          DEC     R1                  ; $$TMP1
```

```
7390 025764  001375                    BNE     2$
7391 025766  005300              3$:   DEC     R0                      ; $$TMP2
7392 025770  000767                    BR      1$
7393 025772  032777  177670  165214  4$:   BIT     #-110,@ML.REG+40        ;                       4480
7394 026000  001426                    BEQ     5$
7395 026002  104455                    TRAP    55                      ;                       4483
7396 026004  000167                    .WORD   167
7397 026006  011060                    .WORD   RH.ERROR
7398 026010  024060                    .WORD   DUMPER
7399 026012  012746  007454            MOV     #PHR.4,-(SP)            ;                       4484
7400 026016  012746  010242            MOV     #FNC.23,-(SP)
7401 026022  012746  010420            MOV     #REG.17,-(SP)
7402 026026  012746  006034            MOV     #THR.FMT,-(SP)
7403 026032  012746  000004            MOV     #4,-(SP)
7404 026036  010600                    MOV     SP,R0                   ; SP,*
7405 026040  104414                    TRAP    14
7406 026042  016700  165532            MOV     ML.LUN,R0               ;                       4485
7407 026046  104451                    TRAP    51
7408 026050  104444                    TRAP    44
7409 026052  062706  000012            ADD     #12,SP                  ;                       4482
7410 026056  005726              5$:   TST     (SP)+                   ;                       4454
7411 026060  012601                    MOV     (SP)+,R1
7412 026062  000207                    RTS     PC
7413
7414                                   ; Routine Size:  59 words
7415                                   ; Maximum stack depth per invocation:  7 words
7420
7421
7425
7426                                   .SBTTL  T1
7430 026064                      T1::
7431 026064  004767  177606      1$:   JSR     PC,$T1                  ;                       4487
7432 026070  104466                    TRAP    66
7433 026072  006000                    ROR     R0
7434 026074  103773                    BLO     1$
7435 026076  000207                    RTS     PC
7436
7437                                   ;ML4                                    14-May-1981 13:48:54    TOPS
7438                                   ;                                       14-May-1981 13:05:08    PA:<
7439
7440                                   ; Routine Size:  6 words
7441                                   ; Maximum stack depth per invocation:  0 words
7446
7447
7448 ;          4490  !<BLF/PAGE>
```

```
7450 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
7451 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (52)
7452
7453 ;          4491  !
7454 ;          4492  BGNTST;
7455 ;          4493
7456 ;          4494  !++
7457 ;          4495  ! TEST NUMBER:  TST 2
7458 ;          4496  !
7459 ;          4497  ! TEST NAME:    MASS BUS HANDSHAKE TEST
7460 ;          4498  !
7461 ;          4499  ! TEST DESCRIPTION:
7462 ;          4500  !       TEST MASS BUS TO UNIBUS COMMUMICATIONS
7463 ;          4501  !       VIA THE CONTROL BUS BY:
7464 ;          4502  !
7465 ;          4503  !       1.  READING RH AND DRIVE REGISTERS AND
7466 ;          4504  !           TEST THE NON EXISTANT DRIVE 'NED'
7467 ;          4505  !           BIT
7468 ;          4506  !
7469 ;          4507  !--
7470 ;          4508
7471 ;          4509  local
7472 ;          4510      SAVE,                                  !TEMPORARY SAVE LOCATION
7473 ;          4511      DODU_FLG;                              !DROP UNIT FLAG
7474 ;          4512
7475 ;          4513  DODU_FLG = ZERO;
7476 ;          4514
7477 ;          4515  incr REG_SEL from 0 to .LST_DUT_REG do     !TEST ALL PRESENT RH REGISTERS
7478 ;          4516      begin
7479 ;          4517      BGNSUB;                                !START OF SCOPE LOOP
7480 ;          4518      CLR_MBUS;
7481 ;          4519      SAVE = ..ML_REG [.REG_SEL, REGISTER_ADD];   !READ THE REGISTER
7482 ;          4520
7483 ;          4521      if .NED IS_SET                         !DID READ CAUSE THE NED BIT TO SET
7484 ;          4522      then
7485 ;          4523          begin                              !REPORT AN ERROR IF SET
7486 ;          4524          ERRDF (120, RH_ERROR, DUMPER);
7487 ;          4525          PRINTB (FIV_FMT, WRD_62, PHR_5, WRD_12, WRD_52, FNC_6);
7488 ;          4526          PRINTB (FMT_11, .ML_REG [.REG_SEL, REGISTER_ADD]);
7489 ;          4527          DODU_FLG = ONE;
7490 ;          4528          end;
7491 ;          4529
7492 ;          4530      ENDSUB;                                !END OF SCOPE LOOP
7493 ;          4531      end;
7494 ;          4532
7495 ;          4533  if .DODU_FLG IS_SET
7496 ;          4534  then
7497 ;          4535      begin
7498 ;          4536      DODU (.ML_LUN);
7499 ;          4537      DOCLN;
7500 ;          4538      end;
7501 ;          4539
7502 ;          4540  ENDTST;
```

```
 7510
 7511                                          .SBTTL  $T2
 7515 026100  004167  156012          $T2:     JSR     R1,$SAVE5                ;                                4489
 7516 026104  005001                           CLR     R1                       ; DODU.FLG                       4513
 7517 026106  016704  165022                   MOV     LST.DUT.REG,R4           ;                                4515
 7518 026112  005002                           CLR     R2                       ; REG.SEL
 7519 026114  000474                           BR      4$
 7520 026116  010200                  1$:      MOV     R2,R0                    ; REG.SEL,*                      4519
 7521 026120  006300                           ASL     R0
 7522 026122  006300                           ASL     R0
 7523 026124  006300                           ASL     R0
 7524 026126  010003                           MOV     R0,R3
 7525 026130  104402                  2$:      TRAP    2                        ;                                4516
 7526 026132  152777  000040  165054           BISB    #40,@ML.REG+40           ;                                4517
 7527 026140  016700  165436                   MOV     ML.DUT,R0
 7528 026144  042700  177770                   BIC     #177770,R0
 7529 026150  142777  000007  165036           BICB    #7,@ML.REG+40
 7530 026156  150077  165032                   BISB    R0,@ML.REG+40
 7531 026162  017305  013154                   MOV     @ML.REG(R3),R5           ; *,SAVE                         4519
 7532 026166  032777  010000  165020           BIT     #10000,@ML.REG+40        ;                                4521
 7533 026174  001440                           BEQ     3$
 7534 026176  104455                           TRAP    55                       ;                                4524
 7535 026200  000170                           .WORD   170
 7536 026202  011060                           .WORD   RH.ERROR
 7537 026204  024060                           .WORD   DUMPER
 7538 026206  012746  010004                   MOV     #FNC.6,-(SP)             ;                                4525
 7539 026212  012746  007066                   MOV     #WRD.52,-(SP)
 7540 026216  012746  006426                   MOV     #WRD.12,-(SP)
 7541 026222  012746  007472                   MOV     #PHR.5,-(SP)
 7542 026226  012746  007210                   MOV     #WRD.62,-(SP)
 7543 026232  012746  006062                   MOV     #FIV.FMT,-(SP)
 7544 026236  012746  000006                   MOV     #6,-(SP)
 7545 026242  010600                           MOV     SP,R0                    ; SP,*
 7546 026244  104414                           TRAP    14
 7547 026246  016316  013154                   MOV     ML.REG(R3),(SP)          ;                                4526
 7548 026252  012746  004750                   MOV     #FMT.11,-(SP)
 7549 026256  012746  000002                   MOV     #2,-(SP)
 7550 026262  010600                           MOV     SP,R0                    ; SP,*
 7551 026264  104414                           TRAP    14
 7552 026266  012701  000001                   MOV     #1,R1                    ; *,DODU.FLG                     4527
 7553 026272  062706  000022                   ADD     #22,SP                   ;                                4523
 7554 026276  104467                  3$:      TRAP    67                       ;                                4528
 7555 026300  006000                           ROR     R0
 7556 026302  103712                           BLO     2$
 7557 026304  005202                           INC     R2                       ; REG.SEL                        4515
 7558 026306  020204                  4$:      CMP     R2,R4                    ; REG.SEL,*
 7559 026310  003702                           BLE     1$
 7560 026312  005301                           DEC     R1                       ; DODU.FLG                       4533
```

```
                                ;ML4                                                    14-May-1981 13:48:54    TOPS
                                ;                                                       14-May-1981 13:05:08    PA:<
                                        BNE     5$
026316  016700  165256                MOV     ML.LUN,R0                       ;                               4536
026322  104451                        TRAP    51
026324  104444                        TRAP    44
026326  000207                5$:     RTS     PC                              ;                               4489

                                ; Routine Size:  76 words
                                ; Maximum stack depth per invocation:  15 words


                                        .SBTTL  T2
026330                          T2::
026330  004767  177544          1$:     JSR     PC,$T2                          ;                               4538
026334  104466                        TRAP    66
026336  006000                        ROR     R0
026340  103773                        BLO     1$
026342  000207                        RTS     PC

                                ; Routine Size:  6 words
                                ; Maximum stack depth per invocation:  0 words


;       4541    !<BLF/PAGE>
```

```
7604 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
7605 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (53)
7606
7607 ;          4542                                        !
7608 ;          4543  BGNTST;
7609 ;          4544
7610 ;          4545  !++
7611 ;          4546  ! TEST NUMBER:  TST 3
7612 ;          4547  !
7613 ;          4548  ! TEST NAME:    DRIVE PRESENT TEST
7614 ;          4549  !
7615 ;          4550  ! TEST DESCRIPTION:
7616 ;          4551  !       THIS TEST READS THE DESIRED SECTOR
7617 ;          4552  !       ADDRESS REGISTER OF THE DRIVE UNDER
7618 ;          4553  !       TEST, DELAYS 100 US, THEN
7619 ;          4554  !       READS THE  NED BIT OF MLCS2
7620 ;          4555  !
7621 ;          4556  !       IF SET, AN ERROR MESSAGE IS
7622 ;          4557  !       PRINTED AND THE UNIT IS DROPPED
7623 ;          4558  !
7624 ;          4559  !--
7625 ;          4560
7626 ;          4561  local
7627 ;          4562      DODU_FLG,                          !DROP UNIT FLAG
7628 ;          4563      SAVE;                              !TEMP STORAGE LOCATION
7629 ;          4564
7630 ;          4565  DODU_FLG = ZERO;
7631 ;          4566  BGNSUB;
7632 ;          4567  CLR_MBUS;
7633 ;          4568  SAVE = .MLDA;                          !READ A DRIVE REGISTER
7634 ;          4569  DELAY (ONE_US);                        !DELAY 1 US
7635 ;          4570
7636 ;          4571  if .NED IS_SET                         !TEST THE NED BIT
7637 ;          4572  then
7638 ;          4573      begin
7639 ;          4574      ERRDF (1, ASYNC, DUMPER);          !IF SET THEN REPORT ERROR AND SET DODU_FLG
7640 ;          4575      PRINTB (ONE_FMT, PHR_3);
7641 ;          4576      DODU_FLG = ONE;
7642 ;          4577      end;
7643 ;          4578
7644 ;          4579  ENDSUB;
7645 ;          4580
7646 ;          4581  if .DODU_FLG IS_SET                    !DROP THIS UNIT IF DODU IS_SET
7647 ;          4582  then
7648 ;          4583      begin
7649 ;          4584      DODU (.ML_LUN);
7650 ;          4585      DOCLN;
7651 ;          4586      end;
7652 ;          4587
7653 ;          4588  ENDTST;
7657
7658                                  .SBTTL  $T3
```

```
                                                                                    14-May-1981 13:48:54    TOPS
                                                                                    14-May-1981 13:05:08    PA:<
                        ;ML4
                        ;
026344  004167  155526  $T3:    JSR     R1,$SAVE4                   ;                                       4540
026350  005746          TST     -(SP)
026352  005003          CLR     R3                                  ; DODU.FLG                              4565
026354  104402  1$:     TRAP    2
026356  152777  000040  164630  BISB    #40,@ML.REG+40              ;                                       4566
026364  016702  165212          MOV     ML.DUT,R2
026370  042702  177770          BIC     #177770,R2
026374  142777  000007  164612  BICB    #7,@ML.REG+40
026402  150277  164606          BISB    R2,@ML.REG+40
026406  017704  164572          MOV     @ML.REG+30,R4               ; *,SAVE                                4568
026412  012701  000001          MOV     #1,R1                       ; *,$$TMP2                              4569
026416  001410          2$:     BEQ     5$
026420  016702  153472          MOV     L$DLY,R2                    ; *,$$TMP1
026424  001403          BEQ     4$
026426  005016          3$:     CLR     (SP)                        ; $$TMP
026430  005302          DEC     R2                                  ; $$TMP1
026432  001375          BNE     3$
026434  005301          4$:     DEC     R1                          ; $$TMP2
026436  000767          BR      2$
026440  032777  010000  164546  5$:     BIT     #10000,@ML.REG+40   ;                                       4571
026446  001420          BEQ     6$
026450  104455          TRAP    55                                  ;                                       4574
026452  000001          .WORD   1
026454  010464          .WORD   ASYNC
026456  024060          .WORD   DUMPER
026460  012746  007422          MOV     #PHR.3,-(SP)                ;                                       4575
026464  012746  003016          MOV     #ONE.FMT,-(SP)
026470  012746  000002          MOV     #2,-(SP)
026474  010600          MOV     SP,R0                               ; SP,*
026476  104414          TRAP    14
026500  012703  000001          MOV     #1,R3                       ; *,DODU.FLG                            4576
026504  062706  000006          ADD     #6,SP                       ;                                       4573
026510  104467          6$:     TRAP    67                          ;                                       4577
026512  006000          ROR     R0
026514  103717          BLO     1$
026516  005303          DEC     R3                                  ; DODU.FLG                              4581
026520  001004          BNE     7$
026522  016700  165052          MOV     ML.LUN,R0                   ;                                       4584
026526  104451          TRAP    51
026530  104444          TRAP    44
026532  005726          7$:     TST     (SP)+                       ;                                       4540
026534  000207          RTS     PC

                        ; Routine Size:  61 words
                        ; Maximum stack depth per invocation:  9 words
```

```
7716 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
7717 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (53)
7718
7722
7723                                        .SBTTL  T3
7727 026536                            T3::
7728 026536  004767  177602            1$:     JSR     PC,$T3                  ;                          4586
7729 026542  104466                            TRAP    66
7730 026544  006000                            ROR     R0
7731 026546  103773                            BLO     1$
7732 026550  000207                            RTS     PC
7733
7734                                   ; Routine Size:  6 words
7735                                   ; Maximum stack depth per invocation:  0 words
7740
7741
7742 ;          4589  !<BLF/PAGE>
```

```
7744 :ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
7745 :                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (54)
7746
7747 :          4590  !
7748 :          4591  BGNTST;
7749 :          4592
7750 :          4593  !++
7751 :          4594  ! TEST NUMBER:  TST 4
7752 :          4595  !
7753 :          4596  ! TEST NAME:    DRIVE SELECTION TEST
7754 :          4597  !
7755 :          4598  ! TEST DESCRIPTION:
7756 :          4599  !       THIS TEST TESTS FOR UNIQUE DRIVE SELECTION BY WRITING THE DRIVE
7757 :          4600  !       UNDER TEST (DUT) DRIVE NUMBER INTO ITS DSA REG.  THEN WRITING THE DRIVE
7758 :          4601  !       NUMBERS OF OTHER DRIVES INTO THEIR RESPECTIVE DSA REGISTERS
7759 :          4602  !       AND READING THE DUT DSA FOR ITS DRIVE NUMBER.
7760 :          4603  !
7761 :          4604  ! IMPLICIT INPUTS:
7762 :          4605  !       ML_DUT
7763 :          4606  !       LOADED DURING THE INITIALIZATION CODE AND CONTAINS THE DRIVE
7764 :          4607  !       NUMBER OF THE DRIVE PRESENTLY BEING TESTED.
7765 :          4608  !--
7766 :          4609
7767 :          4610  local
7768 :          4611      DODU_FLG,                              !DROP UNIT FLAG
7769 :          4612      SAVE;                                  !TEMPORARY SAVE LOCATION
7770 :          4613
7771 :          4614  BGNSUB;
7772 :          4615  CLR_MBUS;
7773 :          4616  DODU_FLG = ZERO;
7774 :          4617  MLDA = .ML_DUT;                            !LOAD THIS DRIVES DRIVE NO. INTO ITS DSA REG
7775 :          4618  SAVE = .MLDA;                              !READ THE REGISTER BACK
7776 :          4619
7777 :          4620  if .SAVE neq .ML_DUT                       !SEE IF DSA HAS DRIVE NUMBER
7778 :          4621  then
7779 :          4622      begin
7780 :          4623      ERRDF (2, INTER, DUMPER);              !ERROR AND EXIT TEST IF DSA NEQ DRIVE NUM
7781 :          4624      PRINTB (SIX_FMT, PHR_4, WRD_12, FNC_3, WRD_37, WRD_13, REG_6);
7782 :          4625      PRINTB (FMT_2, .ML_DUT, .SAVE, (.ML_DUT xor .SAVE));
7783 :          4626      EXIT_TST;
7784 :          4627      end;
7785 :          4628
7786 :          4629  incr DRV_SEL from 0 to 7 do                !WRITE DRV NO OF OTHER DRIVES INTO THEIR RESPECTIVE DSA REG.
7787 :          4630
7788 :          4631      if .DRV_SEL neq .ML_DUT                !SKIP IF .DRV_SEL EQL TO THE DRIVE UNDER TEST (DUT)
7789 :          4632      then
7790 :          4633          begin
7791 :          4634          DRV_NUM = .DRV_SEL;                !SELECT DRIVE TO BE WRITTEN TO
7792 :          4635          MLDA = .DRV_SEL;                   !WRITE DRIVE SEL NO. INTO ITS DSA REG
7793 :          4636          DELAY (ONE_US);                    !DELAY 1 US
7794 :          4637          end;
7795 :          4638
7796 :          4639  DRV_NUM = .ML_DUT;                         !SELECT THE DUT
7797 :          4640  SAVE = .MLDA;                              !READ ITS DSA REG
7798 :          4641
```

```
:ML4                                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
:                                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (54)

:          4642  if .SAVE neq .ML_DUT                                   !SEE IF WRITTING TO OTHER DRIVES CHANGE ITS VALUE
:          4643  then
:          4644      begin
:          4645      ERRDF (3, ASYNC, DUMPER);                         !ERROR AND SET DODU_FLG IF CHANGED
:          4646      PRINTB (THR_FMT, FNC_3, WRD_37, WRD_14);
:          4647      PRINTB (FMT_1, .ML_DUT, .SAVE);
:          4648      DODU_FLG = ONE;
:          4649      end;
:          4650
:          4651  ENDSUB;
:          4652
:          4653  if .DODU_FLG IS_SET                                   !DROP UNIT IF DODU_FLG IS_SET
:          4654  then
:          4655      begin
:          4656      DODU (.ML_LUN);
:          4657      DOCLN;
:          4658      end;
:          4659
:          4660  ENDTST;

                                        .SBTTL  $T4
026552  004167  155340          $T4:    JSR     R1,$SAVE5                 ;                                   4588
026556  005746                          TST     -(SP)
026560  104402                  1$:     TRAP    2                         ;                                   4612
026562  152777  000040  164424          BISB    #40,@ML.REG+40            ;                                   4614
026570  016703  165006                  MOV     ML.DUT,R3
026574  042703  177770                  BIC     #177770,R3
026600  142777  000007  164406          BICB    #7,@ML.REG+40
026606  150377  164402                  BISB    R3,@ML.REG+40
026612  005005                          CLR     R5                        ; DODU.FLG                          4616
026614  016777  164762  164362          MOV     ML.DUT,@ML.REG+30         ;                                   4617
026622  017704  164356                  MOV     @ML.REG+30,R4             ; *,SAVE                            4618
026626  020467  164750                  CMP     R4,ML.DUT                 ; SAVE,*                            4620
026632  001451                          BEQ     2$
026634  104455                          TRAP    55                        ;                                   4623
026636  000002                          .WORD   2
026640  010672                          .WORD   INTER
026642  024060                          .WORD   DUMPER
026644  012746  010316                  MOV     #REG.6,-(SP)              ;                                   4624
026650  012746  006436                  MOV     #WRD.13,-(SP)
026654  012746  006700                  MOV     #WRD.37,-(SP)
026660  012746  007750                  MOV     #FNC.3,-(SP)
026664  012746  006426                  MOV     #WRD.12,-(SP)
026670  012746  007454                  MOV     #PHR.4,-(SP)
026674  012746  006100                  MOV     #SIX.FMT,-(SP)
026700  012746  000007                  MOV     #7,-(SP)
```

```
                                                                                    14-May-1981 13:48:54    TOPS
                                                                                    14-May-1981 13:05:08    PA:<
7856                            ;ML4
7857                            ;
7858
7859 026704  010600                    MOV     SP,R0                   ; SP,*
7860 026706  104414                    TRAP    14
7861 026710  016716  164666            MOV     ML.DUT,(SP)             ;                               4625
7862 026714  010403                    MOV     R4,R3                   ; SAVE,*
7863 026716  041603                    BIC     (SP),R3
7864 026720  040416                    BIC     R4,(SP)                 ; SAVE,*
7865 026722  050316                    BIS     R3,(SP)
7866 026724  010446                    MOV     R4,-(SP)                ; SAVE,*
7867 026726  016746  164650            MOV     ML.DUT,-(SP)
7868 026732  012746  004266            MOV     #FMT.2,-(SP)
7869 026736  012746  000004            MOV     #4,-(SP)
7870 026742  010600                    MOV     SP,R0                   ; SP,*
7871 026744  104414                    TRAP    14
7872 026746  104463                    TRAP    63
7873 026750  062706  000030            ADD     #30,SP                  ;                               4620
7874 026754  000523                    BR      10$                     ;                               4622
7875 026756  005003            2$:     CLR     R3                      ; DRV.SEL                       4629
7876 026760  020367  164616    3$:     CMP     R3,ML.DUT               ; DRV.SEL,*                     4631
7877 026764  001425                    BEQ     7$
7878 026766  010302                    MOV     R3,R2                   ; DRV.SEL,*                     4634
7879 026770  042702  177770            BIC     #177770,R2
7880 026774  142777  000007  164212    BICB    #7,@ML.REG+40
7881 027002  150277  164206            BISB    R2,@ML.REG+40
7882 027006  010377  164172            MOV     R3,@ML.REG+30           ; DRV.SEL,*                     4635
7883 027012  012701  000001            MOV     #1,R1                   ; *,$$TMP2                      4636
7884 027016  001410            4$:     BEQ     7$
7885 027020  016702  153072            MOV     L$DLY,R2                ; *,$$TMP1
7886 027024  001403                    BEQ     6$
7887 027026  005016            5$:     CLR     (SP)                    ; $$TMP
7888 027030  005302                    DEC     R2                      ; $$TMP1
7889 027032  001375                    BNE     5$
7890 027034  005301            6$:     DEC     R1                      ; $$TMP2
7891 027036  000767                    BR      4$
7892 027040  005203            7$:     INC     R3                      ; DRV.SEL                       4629
7893 027042  020327  000007            CMP     R3,#7                   ; DRV.SEL,*
7894 027046  003744                    BLE     3$
7895 027050  016703  164526            MOV     ML.DUT,R3               ;                               4639
7896 027054  042703  177770            BIC     #177770,R3
7897 027060  142777  000007  164126    BICB    #7,@ML.REG+40
7898 027066  150377  164122            BISB    R3,@ML.REG+40
7899 027072  017704  164106            MOV     @ML.REG+30,R4           ; *,SAVE                        4640
7900 027076  020467  164500            CMP     R4,ML.DUT               ; SAVE,*                        4642
7901 027102  001435                    BEQ     8$
7902 027104  104455                    TRAP    55                      ;                               4645
7903 027106  000003                    .WORD   3
7904 027110  010464                    .WORD   ASYNC
7905 027112  024060                    .WORD   DUMPER
7906 027114  012746  006442            MOV     #WRD.14,-(SP)           ;                               4646
7907 027120  012746  006700            MOV     #WRD.37,-(SP)
7908 027124  012746  007750            MOV     #FNC.3,-(SP)
7909 027130  012746  006034            MOV     #THR.FMT,-(SP)
7910 027134  012746  000004            MOV     #4,-(SP)
```

```
                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                        ;                                                       14-May-1981 13:05:08    PA:<
                
027140  010600                  MOV     SP,R0                   ; SP,*
027142  104414                  TRAP    14
027144  010416                  MOV     R4,(SP)                 ; SAVE,*                                4647
027146  016746  164430          MOV     ML.DUT,-(SP)
027152  012746  004222          MOV     #FMT.1,-(SP)
027156  012746  000003          MOV     #3,-(SP)
027162  010600                  MOV     SP,R0                   ; SP,*
027164  104414                  TRAP    14
027166  012705  000001          MOV     #1,R5                   ; *,DODU.FLG                            4648
027172  062706  000020          ADD     #20,SP                  ;                                       4644
027176  104467          8$:     TRAP    67                      ;                                       4649
027200  006000                  ROR     R0
027202  103002                  BHIS    9$
027204  000167  177350          JMP     1$
027210  005305          9$:     DEC     R5                      ; DODU.FLG                              4653
027212  001004                  BNE     10$
027214  016700  164360          MOV     ML.LUN,R0               ;                                       4656
027220  104451                  TRAP    51
027222  104444                  TRAP    44
027224  005726          10$:    TST     (SP)+                   ;                                       4588
027226  000207                  RTS     PC

                        ; Routine Size:  151 words
                        ; Maximum stack depth per invocation:  19 words


                                .SBTTL  T4
027230                  T4::
027230  004767  177316  1$:     JSR     PC,$T4                  ;                                       4658
027234  104466                  TRAP    66
027236  006000                  ROR     R0
027240  103773                  BLO     1$
027242  000207                  RTS     PC

                        ; Routine Size:  6 words
                        ; Maximum stack depth per invocation:  0 words
;               4661    !<BLF/PAGE>
```

```
7968 :ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
7969 :                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (55)
7970
7971 :          4662  !
7972 :          4663  BGNTST;
7973 :          4664
7974 :          4665  !++
7975 :          4666  ! TEST NUMBER:  TST 5
7976 :          4667  !
7977 :          4668  ! TEST NAME:    READ WRITE REG ONES/ZEROES TEST
7978 :          4669  !
7979 :          4670  ! TEST DESCRIPTION:
7980 :          4671  !       THIS TEST WRITES AND READS A DATA PATTERN OF ALL ONES AND ZEROES TO ALL
7981 :          4672  !       OF THE ML11'S READ / WRITE REGISTERS.
7982 :          4673  !
7983 :          4674  !       ROUTINES WRT_REG AND RD_REG ACCEPT ARGUMENTS TO FURTHER SELECT ROUTINES
7984 :          4675  !       WHICH ACTUALLY PERFORMS THE READING AND WRITING OF THE REGISTERS.
7985 :          4676  !
7986 :          4677  !       THE UNIT IS DROPPED ON DETECTED ERRORS.
7987 :          4678  !
7988 :          4679  ! IMPLICIT INPUTS:
7989 :          4680  !       WT_DATA
7990 :          4681  !       LOADED BY READ REGISTER ROUTINES AND CONTAINS THE DATA PATTERN WRITTEN
7991 :          4682  !       TO THE REGISTERS (REPRESENTS GOOD DATA).
7992 :          4683  !
7993 :          4684  !       RD_DATA
7994 :          4685  !       LOADED BY THE READ REGISTER ROUTINES AND CONTAINS THE DATA PATTERN
7995 :          4686  !       READ FROM THE REGISTER (REPRESENTS BAD DATA).
7996 :          4687  !--
7997 :          4688
7998 :          4689  local
7999 :          4690      ERR_FLG,                          !ERROR FLAG PASSED TO ROUTINES
8000 :          4691      TST_PAT,                          !TEST PATTERN
8001 :          4692      index,                            !POINTS TO REGISTER PRESENTLY BEING TESTED
8002 :          4693      DODU_FLG;                         !DROP UNIT FLAG
8003 :          4694
8004 :          4695  DODU_FLG = ZERO;
8005 :          4696  TST_PAT = ONES;                       !LOAD TEST PAT WITH ONES
8006 :          4697
8007 :          4698  incr TWICE from 0 to 1 do             !REPEAT LOOP TWICE
8008 :          4699      begin
8009 :          4700
8010 :          4701      incr REG_SEL from 0 to 7 do       !TEST ELEVEN WRITE/READ REGISTERS
8011 :          4702          begin
8012 :          4703          BGNSUB;
8013 :          4704          CLR_MBUS;
8014 :          4705          WRT_REG (.TST_PAT, .REG_SEL, index);   !WRITE TO THE REGISTER
8015 :          4706          RD_REG (.TST_PAT, .REG_SEL, ERR_FLG);  !READ THE REGISTER
8016 :          4707
8017 :          4708          if .ERR_FLG IS_SET                !SEE IF READ FOUND AN ERROR
8018 :          4709          then
8019 :          4710              begin                         !IF ERROR FLAG IS_SET THEN ERROR AND SET DODU_FLG
8020 :          4711
8021 :          4712              selectone .REG_SEL of         !SELECT WHICH MODULE FAILED
8022 :          4713                  set
8023 :ML4                                                    14-May-1931 13:48:54    TOPS-20 Bliss-16 V2(212)
8024 :                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (55)
```

```
                4714
                4715                    [0 to 3] :
                4716                        ERRDF (4, ASYNC, DUMPER);   !ASYNC MODULE FAILURE
                4717
                4718                    [4 to 5] :
                4719                        ERRDF (4, SYNC, DUMPER);    !SYNC MODULE FAILURE
                4720
                4721                    [6 to 7] :
                4722                        ERRDF (4, ARR_DAT, DUMPER); !ARRAY DATA MODULE FAILURE
                4723                    tes;
                4724
                4725                PRINTB (SIX_FMT, PHR_4, WRD_12, FNC_5, FNC_6, WRD_52, WRD_56);
                4726                PRINTB (FMT_16, .ML_REG [.index, REGISTER_ADD], .WT_DATA, .RD_DATA);
                4727                DODU_FLG = ONE;
                4728                end;
                4729
                4730            ENDSUB;
                4731            end;
                4732
                4733        TST_PAT = not .TST_PAT;                     !REPEAT AGAIN WITH COMPLIMENT DATA
                4734        end;
                4735
                4736    if .DODU_FLG IS_SET                             !DROP THIS UNIT IF THE DODU_FLG IS_SET
                4737    then
                4738        begin
                4739        DODU (.ML_LUN);
                4740        DOCLN;
                4741        end;
                4742
                4743    ENDTST;
```

```
                                    .SBTTL  $T5
027244 004167 154626        $T5:    JSR     R1,$SAVE4           ;                                   4660
027250 024646                       CMP     -(SP),-(SP)
027252 005004                       CLR     R4                  ; DODU.FLG                          4695
027254 012702 177777                MOV     #-1,R2              ; *,TST.PAT                         4696
027260 005001                       CLR     R1                  ; TWICE                             4698
027262 005003               1$:     CLR     R3                  ; REG.SEL                           4701
027264 104402               2$:     TRAP    2                   ;                                   4702
027266 152777 000040 163720         BISB    #40,@ML.REG+40      ;                                   4703
027274 016700 164302                MOV     ML.DUT,R0
027300 042700 177770                BIC     #177770,R0
027304 142777 000007 163702         BICB    #7,@ML.REG+40
027312 150077 163676                BISB    R0,@ML.REG+40
027316 010246                       MOV     R2,-(SP)            ; TST.PAT,*                         4705
027320 010346                       MOV     R3,-(SP)            ; REG.SEL,*
                            ;ML4                                                14-May-1981 13:48:54    TOPS
                            ;                                                   14-May-1981 13:05:08    PA:<
027322 012746 000010                MOV     #10,-(SP)
027326 060616                       ADD     SP,(SP)             ; INDEX,*
027330 004767 173546                JSR     PC,WRT.REG
027334 010216                       MOV     R2,(SP)             ; TST.PAT,*                         4706
027336 010346                       MOV     R3,-(SP)            ; REG.SEL,*
027340 012746 000012                MOV     #12,-(SP)
027344 060616                       ADD     SP,(SP)             ; ERR.FLG,*
```

```
 8088 027346  004767  174116                    JSR     PC,RD.REG
 8089 027352  026627  000012  000001            CMP     12(SP),#1       ; ERR.FLG,*                     4708
 8090 027360  001106                            BNE     6$
 8091 027362  005703                            TST     R3              ; REG.SEL                       4712
 8092 027364  002410                            BLT     3$
 8093 027366  020327  000003                    CMP     R3,#3           ; REG.SEL,*
 8094 027372  003005                            BGT     3$
 8095 027374  104455                            TRAP    55              ;                               4716
 8096 027376  000004                            .WORD   4
 8097 027400  010464                            .WORD   ASYNC
 8098 027402  024060                            .WORD   DUMPER
 8099 027404  000425                            BR      5$              ;                               4712
 8100 027406  020327  000004          3$:       CMP     R3,#4           ; REG.SEL,*
 8101 027412  002410                            BLT     4$
 8102 027414  020327  000005                    CMP     R3,#5           ; REG.SEL,*
 8103 027420  003005                            BGT     4$
 8104 027422  104455                            TRAP    55              ;                               4719
 8105 027424  000004                            .WORD   4
 8106 027426  010526                            .WORD   SYNC
 8107 027430  024060                            .WORD   DUMPER
 8108 027432  000412                            BR      5$              ;                               4712
 8109 027434  020327  000006          4$:       CMP     R3,#6           ; REG.SEL,*
 8110 027440  002407                            BLT     5$
 8111 027442  020327  000007                    CMP     R3,#7           ; REG.SEL,*
 8112 027446  003004                            BGT     5$
 8113 027450  104455                            TRAP    55              ;                               4722
 8114 027452  000004                            .WORD   4
 8115 027454  010570                            .WORD   ARR.DAT
 8116 027456  024060                            .WORD   DUMPER
 8117 027460  012746  007122          5$:       MOV     #WRD.56,-(SP)   ;                               4725
 8118 027464  012746  007066                    MOV     #WRD.52,-(SP)
 8119 027470  012746  010004                    MOV     #FNC.6,-(SP)
 8120 027474  012746  007774                    MOV     #FNC.5,-(SP)
 8121 027500  012746  006426                    MOV     #WRD.12,-(SP)
 8122 027504  012746  007454                    MOV     #PHR.4,-(SP)
 8123 027510  012746  006100                    MOV     #SIX.FMT,-(SP)
 8124 027514  012746  000007                    MOV     #7,-(SP)
 8125 027520  010600                            MOV     SP,R0           ; SP,*
 8126 027522  104414                            TRAP    14
 8127 027524  016716  163400                    MOV     RD.DATA,(SP)    ;                               4726
 8128 027530  016746  163372                    MOV     WT.DATA,-(SP)
 8129 027534  016600  000036                    MOV     36(SP),R0       ; INDEX,*
 8130 027540  006300                            ASL     R0
 8131 027542  006300                            ASL     R0
 8132 027544  006300                            ASL     R0
 8133                               ;ML4                                 14-May-1981 13:48:54    TOPS
 8134                               ;                                    14-May-1981 13:05:08    PA:<
 8135
 8136 027546  016046  013154                    MOV     ML.REG(R0),-(SP)
 8137 027552  012746  005216                    MOV     #FMT.16,-(SP)
 8138 027556  012746  000004                    MOV     #4,-(SP)
 8139 027562  010600                            MOV     SP,R0           ; SP,*
 8140 027564  104414                            TRAP    14
 8141 027566  012704  000001                    MOV     #1,R4           ; *.DODU.FLG                    4727
 8142 027572  062706  000030                    ADD     #30,SP          ;                               4710
 8143 027576  062706  000012          6$:       ADD     #12,SP          ;                               4702
 8144 027602  104467                            TRAP    67              ;                               4728
```

```
                                                                                                                    H 15
 8145 027604  006000                    ROR     R0
 8146 027606  103626                    BLO     2$
 8147 027610  005203                    INC     R3                              ; REG.SEL                       4701
 8148 027612  020327  000007            CMP     R3,#7                           ; REG.SEL,*
 8149 027616  003622                    BLE     2$
 8150 027620  005102                    COM     R2                              ; TST.PAT                       4733
 8151 027622  005201                    INC     R1                              ; TWICE                         4698
 8152 027624  020127  000001            CMP     R1,#1                           ; TWICE,*
 8153 027630  003614                    BLE     1$
 8154 027632  005304                    DEC     R4                              ; DODU.FLG                      4736
 8155 027634  001004                    BNE     7$
 8156 027636  016700  163736            MOV     ML.LUN,R0                       ;                               4739
 8157 027642  104451                    TRAP    51
 8158 027644  104444                    TRAP    44
 8159 027646  022626            7$:     CMP     (SP)+,(SP)+                     ;                               4660
 8160 027650  000207                    RTS     PC
 8161
 8162                           ; Routine Size:  131 words
 8163                           ; Maximum stack depth per invocation:  24 words
 8168
 8169
 8173
 8174                                   .SBTTL  T5
 8178 027652                    T5::
 8179 027652  004767  177366    1$:     JSR     PC,$T5                          ;                               4741
 8180 027656  104466                    TRAP    66
 8181 027660  006000                    ROR     R0
 8182 027662  103773                    BLO     1$
 8183 027664  000207                    RTS     PC
 8184
 8185                           ; Routine Size:  6 words
 8186                           ; Maximum stack depth per invocation:  0 words
 8194
 8195
 8196 ;          4744   !<BLF/PAGE>
```

```
8198 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
8199 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (56)
8200
8201 ;          4745  !
8202 ;          4746  BGNTST;
8203 ;          4747
8204 ;          4748  !++
8205 ;          4749  ! TEST NUMBER:  TST 6
8206 ;          4750  !
8207 ;          4751  ! TEST NAME:    READ WRITE REG SHIFTING ONES/ZEROES TEST
8208 ;          4752  !
8209 ;          4753  ! TEST DESCRIPTION:
8210 ;          4754  !       THIS TEST WRITES AND READS A
8211 ;          4755  !       SHIFTING ONE'S AND SHIFTING ZEROE'S
8212 ;          4756  !       PATTERN TO ALL THE ML11'S
8213 ;          4757  !       READ/WRITE REGISTERS
8214 ;          4758  !
8215 ;          4759  !       ROUTINES WRT_REG AND RD_REG
8216 ;          4760  !       ACCEPT ARGUMENTS TO FURTHER
8217 ;          4761  !       SELECT ROUTINES WHICH ACTUALLY
8218 ;          4762  !       PERFORMS THE READING AND
8219 ;          4763  !       WRITING OF THE REGISTERS.
8220 ;          4764  !
8221 ;          4765  !       THE DRIVE IS DROPPED ON DETECTED
8222 ;          4766  !       ERRORS.
8223 ;          4767  !
8224 ;          4768  ! IMPLICIT INPUTS:
8225 ;          4769  !       WT_DATA
8226 ;          4770  !         LOADED BY READ REGISTER ROUTINES AND
8227 ;          4771  !         CONTAINS THE DATA PATTERN WRITTEN TO THE
8228 ;          4772  !         REGISTERS (REPRESENTS GOOD DATA).
8229 ;          4773  !
8230 ;          4774  !       RD_DATA
8231 ;          4775  !         LOADED BY THE READ REGISTER ROUTINES AND
8232 ;          4776  !         CONTAINS THE DATA PATTERN READ FROM THE
8233 ;          4777  !         REGISTER (REPRESENTS BAD DATA).
8234 ;          4778  !
8235 ;          4779  !
8236 ;          4780  !--
8237 ;          4781
8238 ;          4782  local
8239 ;          4783      ERR_FLG,                          !ERROR FLAG PASSED TO ROUTINE
8240 ;          4784      TST_PAT,                          !TEST PATTERN
8241 ;          4785      index,                            !POINTS TO REG PRESENTLY BEING TESTED
8242 ;          4786      DODU_FLG;                         !DROP UNIT FLAG
8243 ;          4787
8244 ;          4788  DODU_FLG = ZERO;
8245 ;          4789  TST_PAT = ONE;                        !LOAD TST_PAT WITH A 1 IN A FILED OF 0'S
8246 ;          4790
8247 ;          4791  incr SHIFT from 0 to 15 do            !DO SHIFT 16 TIMES
8248 ;          4792      begin
8249 ;          4793
8250 ;          4794      incr TWICE from 0 to 1 do         !REPEAT LOOP TWICE
8251 ;          4795          begin
8252 ;          4796
```

```
8254 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
8255 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (56)
8256
8257 ;          4797        incr REG_SEL from 0 to 7 do          !TEST ELEVEN READ/WRITE REGISTERS
8258 ;          4798            begin
8259 ;          4799            BGNSUB;
8260 ;          4800            CLR_MBUS;
8261 ;          4801            WRT_REG (.TST_PAT, .REG_SEL, index);      !WRITE TO THE RECISTER
8262 ;          4802            RD_REG (.TST_PAT, .REG_SEL, ERR_FLG);     !READ THE REGISTER
8263 ;          4803
8264 ;          4804            if .ERR_FLG IS_SET                        !SEE IF THE READ FOUND AN ERROR
8265 ;          4805            then
8266 ;          4806                begin                                 !IF THE ERROR FLAG IS_SET THEN ERROR
8267 ;          4607
8268 ;          4808                selectone .REG_SEL of                 !FIND WHICH MODULE FAILED
8269 ;          4809                    set
8270 ;          4810
8271 ;          4811                    [0 to 3] :
8272 ;          4812                        ERRDF (5, ASYNC, DUMPER);     !ASYNC MODULE FAILURE
8273 ;          4813
8274 ;          4814                    [4 to 5] :
8275 ;          4815                        ERRDF (5, SYNC, DUMPER);      !SYNC MODULE FAILURE
8276 ;          4816
8277 ;          4817                    [6 to 7] :
8278 ;          4818                        ERRDF (5, ARR_DAT, DUMPER);   !ARRAY DATA MODULE FAILURE
8279 ;          4819                    tes;
8280 ;          4820
8281 ;          4821                PRINTB (SIX_FMT, PHR_4, WRD_12, FNC_5, FNC_6, WRD_52, WRD_56);
8282 ;          4822                PRINTB (FMT_16, .ML_REG [.index, REGISTER_ADD], .WT_DATA, .RD_DATA);
8283 ;          4823                DODU_FLG = ONE;
8284 ;          4824                end;
8285 ;          4825
8286 ;          4826            ENDSUB;
8287 ;          4827            end;
8288 ;          4828
8289 ;          4829        TST_PAT = not .TST_PAT;                   !REPEAT WITH A 0 IN A FIELD OF 1'S
8290 ;          4830        end;
8291 ;          4831
8292 ;          4832    TST_PAT = .TST_PAT^ONE;                       !SHIFT THE 1 IN THE FIELD OF 0'S
8293 ;          4833    end;
8294 ;          4834
8295 ;          4835  if .DODU_FLG IS_SET                             !DROP THIS UNIT IF DODU_FLG IS SET
8296 ;          4836  then
8297 ;          4837      begin
8298 ;          4838      DODU (.ML_LUN);
8299 ;          4839      DOCLN;
8300 ;          4840      end;
8301 ;          4841
8302 ;          4842  ENDTST;
8306
8307                                    .SBTTL  $T6
```

```
 8315 027666  004167  154224          $T6:    JSR     R1,$SAVE5               ;                                       4743
 8316 027672  024646                         CMP     -(SP),-(SP)
 8317 027674  005005                         CLR     R5                      ; DODU.FLG                              4788
 8318 027676  012703  000001                 MOV     #1,R3                   ; *,TST.PAT                             4789
 8319 027702  005001                         CLR     R1                      ; SHIFT                                 4791
 8320 027704  005002                  1$:    CLR     R2                      ; TWICE                                 4794
 8321 027706  005004                  2$:    CLR     R4                      ; REG.SEL                               4797
 8322 027710  104402                  3$:    TRAP    2                       ;                                       4798
 8323 027712  152777  000040  163274         BISB    #40,@ML.REG+40          ;                                       4799
 8324 027720  016700  163656                 MOV     ML.DUT,R0
 8325 027724  042700  177770                 BIC     #177770,R0
 8326 027730  142777  000007  163256         BICB    #7,@ML.REG+40
 8327 027736  150077  163252                 BISB    R0,@ML.REG+40
 8328 027742  010346                         MOV     R3,-(SP)                ; TST.PAT,*                             4801
 8329 027744  010446                         MOV     R4,-(SP)                ; REG.SEL,*
 8330 027746  012746  000010                 MOV     #10,-(SP)
 8331 027752  060616                         ADD     SP,(SP)                 ; INDEX,*
 8332 027754  004767  173122                 JSR     PC,WRT.REG
 8333 027760  010316                         MOV     R3,(SP)                 ; TST.PAT,*                             4802
 8334 027762  010446                         MOV     R4,-(SP)                ; REG.SEL,*
 8335 027764  012746  000012                 MOV     #12,-(SP)
 8336 027770  060616                         ADD     SP,(SP)                 ; ERR.FLG,*
 8337 027772  004767  173472                 JSR     PC,RD.REG
 8338 027776  026627  000012  000001         CMP     12(SP),#1               ; ERR.FLG,*                             4804
 8339 030004  001106                         BNE     7$
 8340 030006  005704                         TST     R4                      ; REG.SEL                               4808
 8341 030010  002410                         BLT     4$
 8342 030012  020427  000003                 CMP     R4,#3                   ; REG.SEL,*
 8343 030016  003005                         BGT     4$
 8344 030020  104455                         TRAP    55                      ;                                       4812
 8345 030022  000005                         .WORD   5
 8346 030024  010464                         .WORD   ASYNC
 8347 030026  024060                         .WORD   DUMPER
 8348 030030  000425                         BR      6$                      ;                                       4808
 8349 030032  020427  000004          4$:    CMP     R4,#4                   ; REG.SEL,*
 8350 030036  002410                         BLT     5$
 8351 030040  020427  000005                 CMP     R4,#5                   ; REG.SEL,*
 8352 030044  003005                         BGT     5$
 8353 030046  104455                         TRAP    55                      ;                                       4815
 8354 030050  000005                         .WORD   5
 8355 030052  010526                         .WORD   SYNC
 8356 030054  024060                         .WORD   DUMPER
 8357 030056  000412                         BR      6$                      ;                                       4808
 8358 030060  020427  000006          5$:    CMP     R4,#6                   ; REG.SEL,*
 8359 030064  002407                         BLT     6$
 8360 030066  020427  000007                 CMP     R4,#7                   ; REG.SEL,*
 8361 030072  003004                         BGT     6$
 8362 030074  104455                         TRAP    55                      ;                                       4818
 8363 030076  000005                         .WORD   5
 8364 030100  010570                         .WORD   ARR.DAT
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                                .WORD   DUMPER
                                6$:     MOV     #WRD.56,-(SP)           ;                                               4821
                                        MOV     #WRD.52,-(SP)
                                        MOV     #FNC.6,-(SP)
                                        MOV     #FNC.5,-(SP)
                                        MOV     #WRD.12,-(SP)
                                        MOV     #PHR.4,-(SP)
                                        MOV     #SIX.FMT,-(SP)
                                        MOV     #7,-(SP)
                                        MOV     SP,R0                   ; SP,*
                                        TRAP    14
                                        MOV     RD.DATA,(SP)            ;                                               4822
                                        MOV     WT.DATA,-(SP)
                                        MOV     36(SP),R0               ; INDEX,*
                                        ASL     R0
                                        ASL     R0
                                        ASL     R0
                                        MOV     ML.REG(R0),-(SP)
                                        MOV     #FMT.16,-(SP)
                                        MOV     #4,-(SP)
                                        MOV     SP,R0                   ; SP,*
                                        TRAP    14
                                        MOV     #1,R5                   ; *,DODU.FLG                                    4823
                                        ADD     #30,SP                  ;                                               4806
                                7$:     ADD     #12,SP                  ;                                               4798
                                        TRAP    67                      ;                                               4824
                                        ROR     R0
                                        BLO     3$
                                        INC     R4                      ; REG.SEL                                       4797
                                        CMP     R4,#7                   ; REG.SEL,*
                                        BLE     3$
                                        COM     R3                      ; TST.PAT                                       4829
                                        INC     R2                      ; TWICE                                         4794
                                        CMP     R2,#1                   ; TWICE,*
                                        BLE     2$
                                        ASL     R3                      ; TST.PAT                                       4832
                                        INC     R1                      ; SHIFT                                         4791
                                        CMP     R1,#17                  ; SHIFT,*
                                        BLE     1$
                                        DEC     R5                      ; DODU.FLG                                      4835
                                        BNE     8$
                                        MOV     ML.LUN,R0               ;                                               4838
                                        TRAP    51
                                        TRAP    44
                                8$:     CMP     (SP)+,(SP)+             ;                                               4743
                                        RTS     PC

                                ; Routine Size:  137 words
                                ; Maximum stack depth per invocation:  25 words
```

```
 8426
 8427
 8431
 8432                                        .SBTTL  T6
 8436 030310                            T6::
 8437 030310  004767  177352            1$:     JSR     PC,$T6                          ;                4840
 8438 030314  104466                            TRAP    66
 8439 030316  006000                            ROR     R0
 8440 030320  103773                            BLO     1$
 8441 030322  000207                            RTS     PC
 8442
 8443                                   ; Routine Size:  6 words
 8444                                   ; Maximum stack depth per invocation:  0 words
 8449
 8450
 8451 ;          4843  !
 8452 ;          4844  !<BLF/PAGE>
```

```
8454 :ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
8455 :                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (57)
8456
8457 :          4845  !
8458 :          4846  BGNTST;
8459 :          4847
8460 :          4848  !++
8461 :          4849  ! TEST NUMBER:  TST 7
8462 :          4850  !
8463 :          4851  ! TEST NAME:    REGISTER INITIALIZATION TEST
8464 :          4852  !
8465 :          4853  ! TEST DESCRIPTION:
8466 :          4854  !       THIS TEST TESTS THE ABILITY OF
8467 :          4855  !       ALL ACCESSIBLE ML11 REGISTERS
8468 :          4856  !       TO CLEAR OUT REGISTER DATA OF
8469 :          4857  !       ONE'S AND ZEROES PATTERN.
8470 :          4858  !
8471 :          4859  !       ROUTINE WRT_REG WRITES A
8472 :          4860  !       PATTERN TO THE SELECTED REGISTER.
8473 :          4861  !
8474 :          4862  !       A MASS BUS CLEAR IS DONE.
8475 :          4863  !
8476 :          4864  !       THEN ROUTINE RD_REG READS THE
8477 :          4865  !       SELECTED REGISTER FOR CLEARED DATA
8478 :          4866  !       THE DRIVE IS DROPPED ON DETECTED ERRORS.
8479 :          4867  !
8480 :          4868  !       THIS TEST WILL ALSO READ THE
8481 :          4869  !       DRIVE TYPE REGISTER FOR ITS
8482 :          4870  !       INITIAL REGISTER VALUE.
8483 :          4871  !
8484 :          4872  ! IMPLICIT INPUTS:
8485 :          4873  !       THIS FLAG IS NEEDED DUE TO THE UNIQUE
8486 :          4874  !       MANNER IN WHICH THESE REGISTERS
8487 :          4875  !       MUST BE READ.
8488 :          4876  !
8489 :          4877  !       RD_DATA
8490 :          4878  !       LOADED BY THE READ REGISTER ROUTINE
8491 :          4879  !       AND CONTAINS THE DATA PATTERN READ FROM
8492 :          4880  !       THE REGISTER (REPRESENTS BAD DATA).
8493 :          4881  !
8494 :          4882  !       DRIVE TYPE
8495 :          4883  !       LOADED DURING THE INITIALIZATION CODE AND
8496 :          4884  !       STORES THE EXPECTED CONTENTS OF THE DRIVE
8497 :          4885  !       TYPE REGISTER.
8498 :          4886  !--
8499 :          4887
8500 :          4888  local
8501 :          4889      TST_PAT,                          !TEST PATTERN
8502 :          4890      ERR_FLG,                          !ERROR FLAG PASSED TO ROUTINE
8503 :          4891      index,                            !POINTS TO REG PRESENTLY BEING TESTED
8504 :          4892      CLR_DATA,                         !STORES CALCULATED REGISTER CLEAR DATA
8505 :          4893      SAVE,                             !TEMPORARY STORAGE LOCATION
8506 :          4894      DODU_FLG;                         !DROP UNIT FLAG
8507 :          4895
8508 :          4896  ML_REG [19, FORCE_HI] = %o'177777';   !CLR DATA FOR MLPD IS ONES
```

```
8510 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
8511 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (57)
8512
8513 ;          4897  DODU_FLG = ZERO;
8514 ;          4898  TST_PAT = ONES;                       !BACKGROUND PATTERN
8515 ;          4899
8516 ;          4900  incr TWICE from 0 to 1 do             !REPEAT LOOP TWICE
8517 ;          4901      begin
8518 ;          4902
8519 ;          4903      incr REG_SEL from 0 to 10 do      !TEST THIRTEEN REGISTERS
8520 ;          4904          begin
8521 ;          4905          BGNSUB;
8522 ;          4906          CLR_MBUS;
8523 ;          4907          WRT_REG (.TST_PAT, .REG_SEL, index);   !WRITE REGISTER WITH BACKGROUND
8524 ;          4908          CLR_DATA = (.AI) or (.IGNORE);         !CALCULATE THE CLEARED DATA PATTERN
8525 ;          4909          CLR_MBUS;                              !CLEAR THE REGISTER
8526 ;          4910          RD_REG (.CLR_DATA, .REG_SEL, ERR_FLG); !READ THE REGISTER FOR THE CLEARED DATA PAT
8527 ;          4911
8528 ;          4912          if .ERR_FLG IS_SET                     !SEE IF READ FOUND AN ERROR
8529 ;          4913          then
8530 ;          4914              begin                              !IF ERROR FLAG IS_SET THEN ERROR AND SET DODU_FLG
8531 ;          4915
8532 ;          4916              selectone .REG_SEL of              !FIND WHICH MODULE FAILED
8533 ;          4917                  set
8534 ;          4918
8535 ;          4919                  [0, 1, 2, 3, 9, 10] :
8536 ;          4920                      ERRDF (6, ASYNC, DUMPER);  .ASYNC MODULE FAILURE
8537 ;          4921
8538 ;          4922                  [4, 5, 8] :
8539 ;          4923                      ERRDF (6, SYNC, DUMPER);   !SYNC MODULE FAILURE
8540 ;          4924
8541 ;          4925                  [6 to 7] :
8542 ;          4926                      ERRDF (6, ARR_DAT, DUMPER); !ARRAY DATA MODULE
8543 ;          4927                  tes;
8544 ;          4928
8545 ;          4929              PRINTB (FIV_FMT, PHR_4, WRD_12, WRD_52, FNC_23, WRD_56);
8546 ;          4930              PRINTB (FMT_16, .ML_REG [.index, REGISTER_ADD], .CLR_DATA, .RD_DATA);
8547 ;          4931              DODU_FLG = ONE;
8548 ;          4932              end;
8549 ;          4933
8550 ;          4934          ENDSUB;
8551 ;          4935          end;
8552 ;          4936
8553 ;          4937      TST_PAT = not .TST_PAT;                !REPEAT WITH COMPLIMENT BACKGROUND PAT
8554 ;          4938      end;
8555 ;          4939
8556 ;          4940  ML_REG [19, FORCE_HI] = ZEROES;          !RESTORE MLPD FORCED_HI
8557 ;          4941  !
8558 ;          4942  !
8559 ;          4943  !  NOW TEST THE DRIVE TYPE REGISTER
8560 ;          4944  !
8561 ;          4945  !
8562 ;          4946
8563 ;          4947  if .MLDT neq .DRIVE_TYPE                 !COMPARE REG CONTENTS TO CALCULATED VALUE
8564 ;          4948  then
```

```
 8566 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
 8567 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BL;.7 (57)
 8568
 8569 ;          4949      begin
 8570 ;          4950      ERRDF (113, ASYNC, DUMPER);
 8571 ;          4951      PRINTB (TWO_FMT, REG_7, PHR_4);
 8572 ;          4952      PRINTB (FMT_2, .DRIVE_TYPE, .SAVE);
 8573 ;          4953      end;
 8574 ;          4954
 8575 ;          4955  if .DODU_FLG IS_SET                    !DROP THIS UNIT IF DODU_FLG IS_SET
 8576 ;          4956  then
 8577 ;          4957      begin
 8578 ;          4958      DODU (.ML_LUN);
 8579 ;          4959      DOCLN;
 8580 ;          4960      end;
 8581 ;          4961
 8582 ;          4962  ENDTST;
 8586
 8587                                          .SBTTL  $T7
 8591 030324  004167  153566          $T7:    JSR     R1,$SAVE5                ;                          4842
 8592 030330  162706  000006                  SUB     #6,SP
 8593 030334  012767  177777  163044          MOV     #-1,ML.REG+232           ;                          4896
 8594 030342  005005                          CLR     R5                       ; DODU.FLG                 4897
 8595 030344  012702  177777                  MOV     #-1,R2                   ; *,TST.PAT                4898
 8596 030350  005001                          CLR     R1                       ; TWICE                    4900
 8597 030352  005004                  1$:     CLR     R4                       ; REG.SEL                  4903
 8598 030354  104402                  2$:     TRAP    2                        ;                          4904
 8599 030356  152777  000040  162630          BISB    #40,@ML.REG+40           ;                          4905
 8600 030364  016703  163212                  MOV     ML.DUT,R3
 8601 030370  042703  177770                  BIC     #177770,R3
 8602 030374  142777  000007  162612          BICB    #7,@ML.REG+40
 8603 030402  150377  162606                  BISB    R3,@ML.REG+40
 8604 030406  010246                          MOV     R2,-(SP)                 ; TST.PAT,*                4907
 8605 030410  010446                          MOV     R4,-(SP)                 ; REG.SEL,*
 8606 030412  012746  000012                  MOV     #12,-(SP)
 8607 030416  060616                          ADD     SP,(SP)                  ; INDEX,*
 8608 030420  004767  172456                  JSR     PC,WRT.REG
 8609 030424  016600  000012                  MOV     12(SP),R0                ; INDEX,*                  4908
 8610 030430  006300                          ASL     R0
 8611 030432  006300                          ASL     R0
 8612 030434  006300                          ASL     R0
 8613 030436  010003                          MOV     R0,R3
 8614 030440  016300  013156                  MOV     ML.REG+2(R3),R0
 8615 030444  056300  013162                  BIS     ML.REG+6(R3),R0
 8616 030450  010066  000006                  MOV     R0,6(SP)                 ; *,CLR.DATA
 8617 030454  152777  000040  162532          BISB    #40,@ML.REG+40
 8618 030462  016700  163114                  MOV     ML.DUT,R0
 8619 030466  042700  177770                  BIC     #177770,R0
 8620 030472  142777  000007  162514          BICB    #7,@ML.REG+40
```

14-May-1981 13:48:54 TOPS
14-May-1981 13:05:08 PA:<

```
8622                                    ;ML4
8623                                    ;
8624
8625 030500  150077  162510                    BISB    R0,@ML.REG+40
8626 030504  016616  000006                    MOV     6(SP),(SP)          ; CLR.DATA,*                    4910
8627 030510  010446                            MOV     R4,-(SP)            ; REG.SEL,*
8628 030512  012746  000014                    MOV     #14,-(SP)
8629 030516  060616                            ADD     SP,(SP)
8630 030520  004767  172744                    JSR     PC,RD.REG           ; ERR.FLG,*
8631 030524  026627  000014  000001            CMP     14(SP),#1           ; ERR.FLG,*                     4912
8632 030532  001110                            BNE     10$
8633 030534  005704                            TST     R4                  ; REG.SEL                       4916
8634 030536  002403                            BLT     3$
8635 030540  020427  000003                    CMP     R4,#3               ; REG.SEL,*
8636 030544  003406                            BLE     4$
8637 030546  020427  000011            3$:     CMP     R4,#11              ; REG.SEL,*
8638 030552  002410                            BLT     5$
8639 030554  020427  000012                    CMP     R4,#12              ; REG.SEL,*
8640 030560  003005                            BGT     5$
8641 030562  104455                    4$:     TRAP    55                  ;                               4920
8642 030564  000006                            .WORD   6
8643 030566  010464                            .WORD   ASYNC
8644 030570  024060                            .WORD   DUMPER
8645 030572  000430                            BR      9$                  ;                               4916
8646 030574  020427  000004            5$:     CMP     R4,#4               ; REG.SEL,*
8647 030600  002403                            BLT     6$
8648 030602  020427  000005                    CMP     R4,#5               ; REG.SEL,*
8649 030606  003403                            BLE     7$
8650 030610  020427  000010            6$:     CMP     R4,#10              ; REG.SEL,*
8651 030614  001005                            BNE     8$
8652 030616  104455                    7$:     TRAP    55                  ;                               4923
8653 030620  000006                            .WORD   6
8654 030622  010526                            .WORD   SYNC
8655 030624  024060                            .WORD   DUMPER
8656 030626  000412                            BR      9$                  ;                               4916
8657 030630  020427  000006            8$:     CMP     R4,#6               ; REG.SEL,*
8658 030634  002407                            BLT     9$
8659 030636  020427  000007                    CMP     R4,#7               ; REG.SEL,*
8660 030642  003004                            BGT     9$
8661 030644  104455                            TRAP    55                  ;                               4926
8662 030646  000006                            .WORD   6
8663 030650  010570                            .WORD   ARR.DAT
8664 030652  024060                            .WORD   DUMPER
8665 030654  012746  007122            9$:     MOV     #WRD.56,-(SP)       ;                               4929
8666 030660  012746  010242                    MOV     #FNC.23,-(SP)
8667 030664  012746  007066                    MOV     #WRD.52,-(SP)
8668 030670  012746  006426                    MOV     #WRD.12,-(SP)
8669 030674  012746  007454                    MOV     #PHR.4,-(SP)
8670 030700  012746  006062                    MOV     #FIV.FMT,-(SP)
8671 030704  012746  000006                    MOV     #6,-(SP)
8672 030710  010600                            MOV     SP,R0               ; SP,*
8673 030712  104414                            TRAP    14
8674 030714  016716  162210                    MOV     RD.DATA,(SP)        ;                               4930
8675 030720  016646  000030                    MOV     30(SP),-(SP)        ; CLR.DATA,*
8676 030724  016346  013154                    MOV     ML.REG(R3),-(SP)
```

```
                                        ;ML4                                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                                       14-May-1981 13:05:08    PA:<
                                        MOV     #FMT.16,-(SP)
                                        MOV     #4,-(SP)
                                        MOV     SP,R0                           ; SP,*
                                        TRAP    14
                                        MOV     #1,R5                           ; *,DODU.FLG                                            4931
                                        ADD     #26,SP                          ;                                                       4914
                                10$:    ADD     #12,SP                          ;                                                       4904
                                        TRAP    67                              ;                                                       4932
                                        ROR     R0
                                        BHIS    12$
                                11$:    JMP     2$
                                12$:    INC     R4                              ; REG.SEL                                               4903
                                        CMP     R4,#12                          ; REG.SEL,*
                                        BLE     11$
                                        COM     R2                              ; TST.PAT                                               4937
                                        INC     R1                              ; TWICE                                                 4900
                                        CMP     R1,#1                           ; TWICE,*
                                        BGT     13$
                                        JMP     1$
                                13$:    CLR     ML.REG+232                      ;                                                       4940
                                        CMP     @ML.REG+130,DRIVE.TYPE          ;                                                       4947
                                        BEQ     14$
                                        TRAP    55                              ;                                                       4950
                                        .WORD   161
                                        .WORD   ASYNC
                                        .WORD   DUMPER
                                        MOV     #PHR.4,-(SP)                    ;                                                       4951
                                        MOV     #REG.7,-(SP)
                                        MOV     #TWO.FMT,-(SP)
                                        MOV     #3,-(SP)
                                        MOV     SP,R0                           ; SP,*
                                        TRAP    14
                                        MOV     R4,(SP)                         ; SAVE,*                                                4952
                                        MOV     DRIVE.TYPE,-(SP)
                                        MOV     #FMT.2,-(SP)
                                        MOV     #3,-(SP)
                                        MOV     SP,R0                           ; SP,*
                                        TRAP    14
                                        ADD     #16,SP                          ;                                                       4949
                                14$:    DEC     R5                              ; DODU.FLG                                              4955
                                        BNE     15$
                                        MOV     ML.LUN,R0                       ;                                                       4958
                                        TRAP    51
                                        TRAP    44
                                15$:    ADD     #6,SP                           ;                                                       4842
                                        RTS     PC

                                ; Routine Size:  198 words
                                ; Maximum stack depth per invocation:  25 words
```

```
 8738
 8739
 8743
 8744                                    .SBTTL  T7
 8748 031140                        T7::
 8749 031140  004767  177160        1$:     JSR     PC,$T7                  ;                4960
 8750 031144  104466                        TRAP    66
 8751 031146  006000                        ROR     R0
 8752 031150  103773                        BLO     1$
 8753 031152  000207                        RTS     PC
 8754
 8755                               ; Routine Size:  6 words
 8756                               ; Maximum stack depth per invocation:  0 words
 8761
 8762
 8763 ;          4963  !<BLF/PAGE>
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (58)

```
8765 :ML4
8766 :
8767
8768 :          4964  !
8769 :          4965  !
8770 :          4966  !
8771 :          4967  BGNTST;
8772 :          4968
8773 :          4969  !++
8774 :          4970  ! TEST NUMBER:  TST 8
8775 :          4971  !
8776 :          4972  ! TEST NAME:    REGISTER SELECTION TEST
8777 :          4973  !
8778 :          4974  ! TEST DESCRIPTION:
8779 :          4975  !       THIS TEST TESTS FOR UNIQUE REGISTER
8780 :          4976  !       SELECTION BY FIRST WRITING A BACKGROUND
8781 :          4977  !       PATTERN INTO ALL READ/WRITE REGISTERS
8782 :          4978  !
8783 :          4979  !       IT THEN WRITES A COMPLIMENT
8784 :          4980  !       BACKGROUND PATTERN INTO ONE REGISTER
8785 :          4981  !       AND READS THE REMAINING UNWRITTEN
8786 :          4982  !       REGISTERS FOR AN UNCHANGED BACKGROUND
8787 :          4983  !       PAT
8788 :          4984  !
8789 :          4985  !       THIS PROCEDURE IS REPEATED UNTIL ALL
8790 :          4986  !       REGISTERS HAVE BEEN WRITTEN WITH A
8791 :          4987  !       COMPLIMENT BACKGROUND PATTERN.
8792 :          4988  !
8793 :          4989  !--
8794 :          4990
8795 :          4991  local
8796 :          4992      DODU_FLG,                       !DROP UNIT FLAG
8797 :          4993      TST_PAT,                        !TEST PATTERN
8798 :          4994      ERR_FLG,                        !ERROR FLAG PASSED TO ROUTINE;
8799 :          4995      index;                          !POINTS TO REGISTER PRESENTLY BEING TESTED.
8800 :          4996
8801 :          4997  BGNSUB;
8802 :          4998  CLR_MBUS;
8803 :          4999  DODU_FLG = ZERO;
8804 :          5000  TST_PAT = %o'125252';               !LOAD TST_PAT WITH ALTERNATE 1'S & 0'S
8805 :          5001  WRT_CS1 (.TST_PAT, 0);              !WRITE A BACKGROUND INTO ALL THE DIRECTLY
8806 :          5002  WRT_ER (.TST_PAT, 6);               !ACCESSABLE READ WRITE REGISTERS
8807 :          5003  WRT_DA (.TST_PAT, 3);
8808 :          5004  WRT_PA (.TST_PAT, 8);
8809 :          5005  WRT_E1 (.TST_PAT, 13);
8810 :          5006  WRT_E2 (.TST_PAT, 14);
8811 :          5007
8812 :          5008  incr CNT_1 from 0 to 4 do           !WRITE A COMPLIMENT PATTERN INTO ONE REGISTER
8813 :          5009      begin
8814 :          5010      TST_PAT = not .TST_PAT;         !GENERATE THE COMPLIMENT PAT
8815 :          5011
8816 :          5012      case .CNT_1 from 0 to 4 of      !SELECT THE REGISTER TO WRITE INTO
8817 :          5013          set
8818 :          5014
8819 :          5015          [0] :
```

```
8821 ;ML4
8822 ;
8823
8824 ;          5016            WRT_CS1 (.TST_PAT, 0);              !FIRST PASS WRITE COMP PAT TO MLCS1
8825 ;          5017
8826 ;          5018        [1] :
8827 ;          5019            WRT_ER (.TST_PAT, 6);               !SECOND PASS WRITE COMP PAT TO MLER
8828 ;          5020
8829 ;          5021        [2] :
8830 ;          5022            WRT_DA (.TST_PAT, 3);               !THIRD PASS WRITE COMP PAT TO MLDA
8831 ;          5023
8832 ;          5024        [3] :
8833 ;          5025            WRT_PA (.TST_PAT, 8);               !FORTH PASS WRITE COMP PAT TO MLPA
8834 ;          5026
8835 ;          5027        [4] :
8836 ;          5028            WRT_E1 (.TST_PAT, 13);              !FIFTH PASS WRITE COMP PAT TO MLE1
8837 ;          5029        tes;
8838 ;          5030
8839 ;          5031    TST_PAT = not .TST_PAT;                     !COMPLIMENT TST_PAT BACK TO BACKGROUND
8840 ;          5032
8841 ;          5033    incr CNT_2 from .CNT_1 + 1 to 5 do          !NOW READ THE REMAINING UNWRITTEN REGISTERS FOR AN
8842 ;          5034        begin                                   !UNCHANGED BACKGROUND
8843 ;          5035
8844 ;          5036        case .CNT_2 from 0 to 5 of              !SELECT THE REGISTER TO READ
8845 ;          5037            set
8846 ;          5038
8847 ;          5039            [0] :
8848 ;          5040                RD_CS1 (.TST_PAT, index = 0, ERR_FLG);  !READ MLCS1
8849 ;          5041
8850 ;          5042            [1] :
8851 ;          5043                RD_ER (.TST_PAT, index = 6, ERR_FLG);   !READ MLER
8852 ;          5044
8853 ;          5045            [2] :
8854 ;          5046                RD_DA (.TST_PAT, index = 3, ERR_FLG);   !READ MLDA
8855 ;          5047
8856 ;          5048            [3] :
8857 ;          5049                RD_PA (.TST_PAT, index = 8, ERR_FLG);   !READ MLPA
8858 ;          5050
8859 ;          5051            [4] :
8860 ;          5052                RD_E1 (.TST_PAT, index = 13, ERR_FLG);  !READ MLE1
8861 ;          5053
8862 ;          5054            [5] :
8863 ;          5055                RD_E2 (.TST_PAT, index = 14, ERR_FLG);  !READ MLE2
8864 ;          5056            tes;
8865 ;          5057
8866 ;          5058        if .ERR_FLG IS_SET                      !SEE IF READ FOUND AN ERROR
8867 ;          5059        then
8868 ;          5060            begin
8869 ;          5061
8870 ;          5062            selectone .CNT_2 of
8871 ;          5063                set
8872 ;          5064
8873 ;          5065                [0 to 2] :
8874 ;          5066                    ERRDF (110, ASYNC, DUMPER); !ASYNC MODULE FAILURE
8875 ;          5067
```

```
 8877 ;ML4                                                                  14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
 8878 ;                                                                     14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (58)
 8879
 8880 ;          5068                [3 to 5] :
 8881 ;          5069                    ERRDF (110, ARR_DAT, DUMPER);       !ARRAY DATA MODULE FAILURE
 8882 ;          5070                tes;
 8883 ;          5071
 8884 ;          5072            PRINTB (THR_FMT, WRD_38, WRD_37, WRD_10);
 8885 ;          5073            PRINTB (FMT_16, .ML_REG [.index, REGISTER_ADD], .WT_DATA, .RD_DATA);
 8886 ;          5074            DODU_FLG = ONE;
 8887 ;          5075            end;
 8888 ;          5076
 8889 ;          5077        end;
 8890 ;          5078
 8891 ;          5079    end;
 8892 ;          5080
 8893 ;          5081  ENDSUB;
 8894 ;          5082
 8895 ;          5083  if .DODU_FLG IS_SET                                  !DROP THIS UNIT IF DODU_FLG IS_SET
 8896 ;          5084  then
 8897 ;          5085      begin
 8898 ;          5086      DODU (.ML_LUN);
 8899 ;          5087      DOCLN;
 8900 ;          5088      end;
 8901 ;          5089
 8902 ;          5090  ENDTST;
 8906
 8907                                    .SBTTL  $T8
 8911 031154  004167  152736          $T8:    JSR     R1,$SAVE5                   ;                           4962
 8912 031160  005746                          TST     -(SP)
 8913 031162  104402                  1$:     TRAP    2                           ;                           4995
 8914 031164  152777  000040  162022          BISB    #40,@ML.REG+40              ;                           4997
 8915 031172  016703  162404                  MOV     ML.DUT,R3
 8916 031176  042703  177770                  BIC     #177770,R3
 8917 031202  142777  000007  162004          BICB    #7,@ML.REG+40
 8918 031210  150377  162000                  BISB    R3,@ML.REG+40
 8919 031214  005005                          CLR     R5                          ; DODU.FLG                  4999
 8920 031216  012704  125252                  MOV     #-52526,R4                  ; *,TST.PAT                 5000
 8921 031222  010446                          MOV     R4,-(SP)                    ; TST.PAT,*                 5001
 8922 031224  005046                          CLR     -(SP)
 8923 031226  004767  166236                  JSR     PC,WRT.CS1
 8924 031232  010416                          MOV     R4,(SP)                     ; TST.PAT,*                 5002
 8925 031234  012746  000006                  MOV     #6,-(SP)
 8926 031240  004767  166402                  JSR     PC,WRT.ER
 8927 031244  010416                          MOV     R4,(SP)                     ; TST.PAT,*                 5003
 8928 031246  012746  000003                  MOV     #3,-(SP)
 8929 031252  004767  166546                  JSR     PC,WRT.DA
 8930 031256  010416                          MOV     R4,(SP)                     ; TST.PAT,*                 5004
 8931 031260  012746  000010                  MOV     #10,-(SP)
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                        
8936 031264  004767  167070                    JSR     PC,WRT.PA
8937 031270  010416                            MOV     R4,(SP)          ; TST.PAT,*                                5005
8938 031272  012746  000015                    MOV     #15,-(SP)
8939 031276  004767  167264                    JSR     PC,WRT.E1
8940 031302  010416                            MOV     R4,(SP)          ; TST.PAT,*                                5006
8941 031304  012746  000016                    MOV     #16,-(SP)
8942 031310  004767  167460                    JSR     PC,WRT.E2
8943 031314  005001                            CLR     R1               ; CNT.1                                    5008
8944 031316  005104                2$:         COM     R4               ; TST.PAT                                  5010
8945 031320  010103                            MOV     R1,R3            ; CNT.1,*                                  5012
8946 031322  006303                            ASL     R3
8947 031324  066307  031330                    ADD     3$(R3),PC
8948 031330  000012                3$:         .WORD   4$-3$
8949 031332  000024                            .WORD   5$-3$
8950 031334  000040                            .WORD   6$-3$
8951 031336  000054                            .WORD   7$-3$
8952 031340  000070                            .WORD   8$-3$
8953 031342  010446                4$:         MOV     R4,-(SP)         ; TST.PAT,*                                5016
8954 031344  005046                            CLR     -(SP)
8955 031346  004767  166116                    JSR     PC,WRT.CS1
8956 031352  000427                            BR      9$               ;                                          5012
8957 031354  010446                5$:         MOV     R4,-(SP)         ; TST.PAT,*                                5019
8958 031356  012746  000006                    MOV     #6,-(SP)
8959 031362  004767  166260                    JSR     PC,WRT.ER
8960 031366  000421                            BR      9$               ;                                          5012
8961 031370  010446                6$:         MOV     R4,-(SP)         ; TST.PAT,*                                5022
8962 031372  012746  000003                    MOV     #3,-(SP)
8963 031376  004767  166422                    JSR     PC,WRT.DA
8964 031402  000413                            BR      9$               ;                                          5012
8965 031404  010446                7$:         MOV     R4,-(SP)         ; TST.PAT,*                                5025
8966 031406  012746  000010                    MOV     #10,-(SP)
8967 031412  004767  166742                    JSR     PC,WRT.PA
8968 031416  000405                            BR      9$               ;                                          5012
8969 031420  010446                8$:         MOV     R4,-(SP)         ; TST.PAT,*                                5028
8970 031422  012746  000015                    MOV     #15,-(SP)
8971 031426  004767  167134                    JSR     PC,WRT.E1
8972 031432  005104                9$:         COM     R4               ; TST.PAT                                  5031
8973 031434  010103                            MOV     R1,R3            ; CNT.1,CNT.2                              5033
8974 031436  000576                            BR      22$
8975 031440  010300                10$:        MOV     R3,R0            ; CNT.2,*                                  5036
8976 031442  006300                            ASL     R0
8977 031444  066007  031450                    ADD     11$(R0),PC
8978 031450  000014                11$:        .WORD   12$-11$
8979 031452  000036                            .WORD   13$-11$
8980 031454  000062                            .WORD   14$-11$
8981 031456  000106                            .WORD   15$-11$
8982 031460  000132                            .WORD   16$-11$
8983 031462  000156                            .WORD   17$-11$
8984 031464  010446                12$:        MOV     R4,-(SP)         ; TST.PAT,*                                5040
8985 031466  005002                            CLR     R2               ; INDEX
8986 031470  005046                            CLR     -(SP)
8987 031472  012746  000030                    MOV     #30,-(SP)
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                                ADD     SP,(SP)                 ; ERR.FLG,*
                                                JSR     PC,RD.CS1
                                                BR      18$                     ;                                       5036
                                        13$:    MOV     R4,-(SP)                ; TST.PAT,*                             5043
                                                MOV     #6,R2                   ; *,INDEX
                                                MOV     R2,-(SP)                ; INDEX,*
                                                MOV     #30,-(SP)
                                                ADD     SP,(SP)                 ; ERR.FLG,*
                                                JSR     PC,RD.ER
                                                BR      18$                     ;                                       5036
                                        14$:    MOV     R4,-(SP)                ; TST.PAT,*                             5046
                                                MOV     #3,R2                   ; *,INDEX
                                                MOV     R2,-(SP)                ; INDEX,*
                                                MOV     #30,-(SP)
                                                ADD     SP,(SP)                 ; ERR.FLG,*
                                                JSR     PC,RD.DA
                                                BR      18$                     ;                                       5036
                                        15$:    MOV     R4,-(SP)                ; TST.PAT,*                             5049
                                                MOV     #10,R2                  ; *,INDEX
                                                MOV     R2,-(SP)                ; INDEX,*
                                                MOV     #30,-(SP)
                                                ADD     SP,(SP)                 ; ERR.FLG,*
                                                JSR     PC,RD.PA
                                                BR      18$                     ;                                       5036
                                        16$:    MOV     R4,-(SP)                ; TST.PAT,*                             5052
                                                MOV     #15,R2                  ; *,INDEX
                                                MOV     R2,-(SP)                ; INDEX,*
                                                MOV     #30,-(SP)
                                                ADD     SP,(SP)                 ; ERR.FLG,*
                                                JSR     PC,RD.E1
                                                BR      18$                     ;                                       5036
                                        17$:    MOV     R4,-(SP)                ; TST.PAT,*                             5055
                                                MOV     #16,R2                  ; *,INDEX
                                                MOV     R2,-(SP)                ; INDEX,*
                                                MOV     #30,-(SP)
                                                ADD     SP,(SP)                 ; ERR.FLG,*
                                                JSR     PC,RD.E2
                                18$:    CMP     30(SP),#1               ; ERR.FLG,*                             5058
                                                BNE     21$
                                                TST     R3                      ; CNT.2                                 5062
                                                BLT     19$
                                                CMP     R3,#2                   ; CNT.2,*
                                                BGT     19$
                                                TRAP    55                      ;                                       5066
                                                .WORD   156
                                                .WORD   ASYNC
                                                .WORD   DUMPER
                                                BR      20$                     ;                                       5062
                                        19$:    CMP     R3,#3                   ; CNT.2,*
                                                BLT     20$
                                                CMP     R3,#5                   ; CNT.2,*
                                                BGT     20$
```

```
 9045                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
 9046                                   ;                                                       14-May-1981 13:05:08    PA:<
 9047
 9048 031720  104455                            TRAP    55                      ;                                       5069
 9049 031722  000156                            .WORD   156
 9050 031724  010570                            .WORD   ARR.DAT
 9051 031726  024060                            .WORD   DUMPER
 9052 031730  012746  006406        20$:        MOV     #WRD.10,-(SP)           ;                                       5072
 9053 031734  012746  006700                    MOV     #WRD.37,-(SP)
 9054 031740  012746  006710                    MOV     #WRD.38,-(SP)
 9055 031744  012746  006034                    MOV     #THR.FMT,-(SP)
 9056 031750  012746  000004                    MOV     #4,-(SP)
 9057 031754  010600                            MOV     SP,R0                   ; SP,*
 9058 031756  104414                            TRAP    14
 9059 031760  016716  161144                    MOV     RD.DATA,(SP)            ;                                       5073
 9060 031764  016746  161136                    MOV     WT.DATA,-(SP)
 9061 031770  010200                            MOV     R2,R0                   ; INDEX,*
 9062 031772  006300                            ASL     R0
 9063 031774  006300                            ASL     R0
 9064 031776  006300                            ASL     R0
 9065 032000  016046  013154                    MOV     ML.REG(R0),-(SP)
 9066 032004  012746  005216                    MOV     #FMT.16,-(SP)
 9067 032010  012746  000004                    MOV     #4,-(SP)
 9068 032014  010600                            MOV     SP,R0                   ; SP,*
 9069 032016  104414                            TRAP    14
 9070 032020  012705  000001                    MOV     #1,R5                   ; *,DODU.FLG                            5074
 9071 032024  062706  000022                    ADD     #22,SP                  ;                                       5060
 9072 032030  062706  000006        21$:        ADD     #6,SP                   ;                                       5034
 9073 032034  005203                22$:        INC     R3                      ; CNT.2                                 5033
 9074 032036  020327  000005                    CMP     R3,#5                   ; CNT.2,*
 9075 032042  003002                            BGT     23$
 9076 032044  000167  177370                    JMP     10$
 9077 032050  022626                23$:        CMP     (SP)+,(SP)+             ;                                       5009
 9078 032052  005201                            INC     R1                      ; CNT.1                                 5008
 9079 032054  020127  000004                    CMP     R1,#4                   ; CNT.1,*
 9080 032060  003002                            BGT     24$
 9081 032062  000167  177230                    JMP     2$
 9082 032066  062706  000016        24$:        ADD     #16,SP                  ;                                       4995
 9083 032072  104467                            TRAP    67                      ;                                       5079
 9084 032074  006000                            ROR     R0
 9085 032076  103002                            BHIS    25$
 9086 032100  000167  177056                    JMP     1$
 9087 032104  005305                25$:        DEC     R5                      ; DODU.FLG                              5083
 9088 032106  001004                            BNE     26$
 9089 032110  016700  161464                    MOV     ML.LUN,R0               ;                                       5086
 9090 032114  104451                            TRAP    51
 9091 032116  104444                            TRAP    44
 9092 032120  005726                26$:        TST     (SP)+                   ;                                       4962
 9093 032122  000207                            RTS     PC
 9094
 9095                                   ; Routine Size:  244 words
 9096                                   ; Maximum stack depth per invocation:  28 words
```

```
 9105
 9106
 9110
 9111                                    .SBTTL  T8
 9115 032124                        T8::
 9116 032124  004767  177024        1$:     JSR     PC,$T8                          ;                5088
 9117 032130  104466                        TRAP    66
 9118 032132  006000                        ROR     R0
 9119 032134  103773                        BLO     1$
 9120 032136  000207                        RTS     PC
 9121
 9122                                ; Routine Size:  6 words
 9123                                ; Maximum stack depth per invocation:  0 words
 9128
 9129
 9130 ;          5091  !<BLF/PAGE>
```

```
 9132 :ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
 9133 :                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (59)
 9134
 9135 :          5092  !
 9136 :          5093  !
 9137 :          5094  BGNTST;
 9138 :          5095
 9139 :          5096  !++
 9140 :          5097  ! TEST NUMBER:  TST 9
 9141 :          5098  !
 9142 :          5099  ! TEST NAME:    PRINT SERIAL NUMBER
 9143 :          5100  !
 9144 :          5101  ! TEST DESCRIPTION:
 9145 :          5102  !       UPON A YES RESPONSE TO THE
 9146 :          5103  !       SOFTWARE QUESTION 'PRINT SERIAL NO?
 9147 :          5104  !       PRINT OUT THE DRIVE UNDER TEST
 9148 :          5105  !       SERIAL NUMBER.
 9149 :          5106  !
 9150 :          5107  !--
 9151 :          5108
 9152 :          5109  external
 9153 :          5110      PRSN;                                !LOCATION WHERE ANSWER TO SW QUESTION IS STORED
 9154 :          5111
 9155 :          5112  local
 9156 :          5113      D3,                                  !STORES DIGIT 3 OF SERIAL NUMBER
 9157 :          5114      D2,                                  !STORES DIGIT 2 OF SERIAL NUMBER
 9158 :          5115      D1,                                  !STORES DIGIT 1 OF SERIAL NUMBER
 9159 :          5116      D0;                                  !STORES DIGIT 0 OF SERIAL NUMBER
 9160 :          5117
 9161 :          5118  if .PRSN IS_SET                         !SEE IF WE PRINT THE SERIAL NUMBER
 9162 :          5119  then
 9163 :          5120      begin                                !PRINT THE SERIAL NUMBER IF THE REPLY WAS YES
 9164 :          5121      D3 = .SN3;                           !LOAD DIGIT 3 OF SN INTO D3
 9165 :          5122      D2 = .SN2;                           !LOAD DIGIT 2 OF SN INTO D2
 9166 :          5123      D1 = .SN1;                           !LOAD DIGIT 1 OF SN INTO D1
 9167 :          5124      D0 = .SN0;                           !LOAD DIGIT 0 OF SN INTO D0
 9168 :          5125
 9169 :          5126      if ((.D3 gtr 9) or (.D2 gtr 9) or (.D1 gtr 9) or (.D0 gtr 9))
 9170 :          5127                                           !SEE IF DIGITS ARE TO BIG FOR COVERSION
 9171 :          5128      then
 9172 :          5129          PRINTB (FMT_25, PHR_15, .MLSN)   !PRINT OCTAL SN IF TO BIG
 9173 :          5130      else
 9174 :          5131          PRINTB (FMT_26, PHR_15, .D3, .D2, .D1, .D0);    !ELSE PRINT DECIMAL SERIAL NUMBER
 9175 :          5132
 9176 :          5133      end;
 9177 :          5134
 9178 :          5135  ENDTST;
 9182
 9183                                  .GLOBL  PRSN
 9184
 9185
 9186                                  .SBTTL  $T9
```

```
 9188                                   ;ML4                                                                    14-May-1981 13:48:54    TOPS
 9189                                   :                                                                       14-May-1981 13:05:08    PA:<
 9190
 9194 032140  004167  151714            ST9:    JSR     R1,$SAVE3                       ;                                               5090
 9195 032144  026727  150226  000001            CMP     PRSN,#1                         ;                                               5118
 9196 032152  001101                            BNE     4$
 9197 032154  017703  161134                    MOV     @ML.REG+140,R3                  ; *,D3                                          5121
 9198 032160  006203                            ASR     R3                              ; D3
 9199 032162  006203                            ASR     R3                              ; D3
 9200 032164  006203                            ASR     R3                              ; D3
 9201 032166  006203                            ASR     R3                              ; D3
 9202 032170  000303                            SWAB    R3                              ; D3
 9203 032172  042703  177760                    BIC     #177760,R3                      ; *,D3
 9204 032176  017700  161112                    MOV     @ML.REG+140,R0                  ; *,D2                                          5122
 9205 032202  000300                            SWAB    R0                              ; D2
 9206 032204  042700  177760                    BIC     #177760,R0                      ; *,D2
 9207 032210  117701  161100                    MOVB    @ML.REG+140,R1                  ; *,D1                                          5123
 9208 032214  006201                            ASR     R1                              ; D1
 9209 032216  006201                            ASR     R1                              ; D1
 9210 032220  006201                            ASR     R1                              ; D1
 9211 032222  006201                            ASR     R1                              ; D1
 9212 032224  042701  177760                    BIC     #177760,R1                      ; *,D1
 9213 032230  117702  161060                    MOVB    @ML.REG+140,R2                  ; *,D0                                          5124
 9214 032234  042702  177760                    BIC     #177760,R2                      ; *,D0
 9215 032240  020327  000011                    CMP     R3,#11                          ; D3,*                                          5126
 9216 032244  003011                            BGT     1$
 9217 032246  020027  000011                    CMP     R0,#11                          ; D2,*
 9218 032252  003006                            BGT     1$
 9219 032254  020127  000011                    CMP     R1,#11                          ; D1,*
 9220 032260  003003                            BGT     1$
 9221 032262  020227  000011                    CMP     R2,#11                          ; D0,*
 9222 032266  003413                            BLE     2$
 9223 032270  017746  161020            1$:     MOV     @ML.REG+140,-(SP)               ;                                               5129
 9224 032274  012746  007714                    MOV     #PHR.15,-(SP)
 9225 032300  012746  005746                    MOV     #FMT.25,-(SP)
 9226 032304  012746  000003                    MOV     #3,-(SP)
 9227 032310  010600                            MOV     SP,R0                           ; SP,*
 9228 032312  104414                            TRAP    14
 9229 032314  000416                            BR      3$                              ;                                               5126
 9230 032316  010246            2$:             MOV     R2,-(SP)                        ; D0,*                                          5131
 9231 032320  010146                            MOV     R1,-(SP)                        ; D1,*
 9232 032322  010046                            MOV     R0,-(SP)                        ; D2,*
 9233 032324  010346                            MOV     R3,-(SP)                        ; D3,*
 9234 032326  012746  007714                    MOV     #PHR.15,-(SP)
 9235 032332  012746  005764                    MOV     #FMT.26,-(SP)
 9236 032336  012746  000006                    MOV     #6,-(SP)
 9237 032342  010600                            MOV     SP,R0                           ; SP,*
 9238 032344  104414                            TRAP    14
 9239 032346  062706  000006                    ADD     #6,SP
 9240 032352  062706  000010            3$:     ADD     #10,SP                          ;                                               5120
 9241 032356  000207                    4$:     RTS     PC                              ;                                               5090
 9242
```

```
 9244                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
 9245                                   ;                                                       14-May-1981 13:05:08    PA:<
 9246
 9247                                   ; Routine Size:  72 words
 9248                                   ; Maximum stack depth per invocation:  11 words
 9253
 9254
 9258
 9259                                           .SBTTL  T9
 9263 032360                           T9::
 9264 032360  004767  177554           1$:      JSR     PC,$T9                          ;       5133
 9265 032364  104466                            TRAP    66
 9266 032366  006000                            ROR     R0
 9267 032370  103773                            BLO     1$
 9268 032372  000207                            RTS     PC
 9269
 9270                                   ; Routine Size:  6 words
 9271                                   ; Maximum stack depth per invocation:  0 words
 9276
 9277
 9278 ;          5136  !<BLF/PAGE>
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (60)

```
9280 ;ML4
9281 ;
9282
9283 ;          5137  !
9284 ;          5138  !
9285 ;          5139  BGNTST;
9286 ;          5140
9287 ;          5141  !++
9288 ;          5142  ! TEST NUMBER:  TST 10
9289 ;          5143  !
9290 ;          5144  ! TEST NAME:    C-BUS PARITY TEST
9291 ;          5145  !
9292 ;          5146  ! TEST DESCRIPTION:
9293 ;          5147  !       TEST THE CONTROL BUS PARITY
9294 ;          5148  !       DETECTION AND GENERATING BY:
9295 ;          5149  !
9296 ;          5150  !       1.  WRITING BAD PARITY TO DEVICE
9297 ;          5151  !           AND TEST CPAR SET.
9298 ;          5152  !
9299 ;          5153  !       2.  WRITING GOOD PARITY TO DEVICE
9300 ;          5154  !           AND TEST CPAR CLR.
9301 ;          5155  !
9302 ;          5156  !       3.  READING DEVICE AND TEST GOOD
9303 ;          5157  !           PARITY GENERATION BY READING
9304 ;          5158  !           MCPE CLR.
9305 ;          5159  !
9306 ;          5160  !--
9307 ;          5161
9308 ;          5162  local
9309 ;          5163      SAVE,                                   !TEMPORARY SAVE LOCATION
9310 ;          5164      TST_PAT;                                !TEST PATTERN
9311 ;          5165
9312 ;          5166  if .PAR_DIS IS_NOT_SET                      !SEE IF PARITY IS DISABLED
9313 ;          5167  then
9314 ;          5168      begin                                   !BEGIN IF PARITY IS ENABLE
9315 ;          5169      TST_PAT = %o'125252';                   !ALTERNATING 1, 0 PATTERN
9316 ;          5170
9317 ;          5171      incr TWICE from 1 to 2 do               !REPEAT LOOP TWICE
9318 ;          5172          begin
9319 ;          5173          CLR_MBUS;
9320 ;          5174          PAT = ONE;                          !GENERATE EVEN PARITY BY SETTING THE PAT BIT
9321 ;          5175          MLDA = .TST_PAT;                    !WRITE BAD PARITY ON CONTROL BUS
9322 ;          5176
9323 ;          5177          if .CPAR IS_NOT_SET                 !READ CPAR BIT SET
9324 ;          5178          then
9325 ;          5179              begin
9326 ;          5180              ERRDF (7, ASYNC, DUMPER);       !ERROR IF NOT SET
9327 ;          5181              PRINTB (THR_FMT, WRD_5, WRD_7, WRD_9);
9328 ;          5182              end;
9329 ;          5183
9330 ;          5184          CLR_MBUS;                           !CLEAR OUT PAT BIT
9331 ;          5185          MLDA = .TST_PAT;                    !WRITE ODD PARITY CONTROL BUS
9332 ;          5186
9333 ;          5187          if .CPAR IS_SET                     !READ CPAR BIT CLEARED
9334 ;          5188          then
```

```
 9336 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
 9337 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (60)
 9338
 9339 ;          5189            begin
 9340 ;          5190            ERRDF (8, ASYNC, DUMPER);               !ERROR IF SET
 9341 ;          5191            PRINTB (THR_FMT, WRD_6, WRD_7, WRD_9);
 9342 ;          5192            end;
 9343 ;          5193
 9344 ;          5194        CLR_MBUS;
 9345 ;          5195        SAVE = .MLDA;                           !READ A REGISTER FROM DUT
 9346 ;          5196
 9347 ;          5197        if .MCPE IS_SET                         !SEE IF GENERATED GOOD PARITY
 9348 ;          5198        then
 9349 ;          5199            begin
 9350 ;          5200            ERRDF (9, ASYNC, DUMPER);           !ERROR IF MCPE IS_SET
 9351 ;          5201            PRINTB (THR_FMT, WRD_6, WRD_7, WRD_8);
 9352 ;          5202            end;
 9353 ;          5203
 9354 ;          5204        TST_PAT = .TST_PAT^ONE;                 !REPEAT WITH SHIFTED DATA
 9355 ;          5205        end;
 9356 ;          5206
 9357 ;          5207    end
 9358 ;          5208 else
 9359 ;          5209    PRINTB (TWO_FMT, WRD_7, WRD_53);            !JUST PRINT MESSAGE IF PARITY IS DISABLED
 9360 ;          5210
 9361 ;          5211 ENDTST;
 9365
 9366                                    .SBTTL  $T10
 9370 032374 004167 151460          $T10:  JSR     R1,$SAVE3        ;                       5135
 9371 032400 005767 157162                 TST     PAR.DIS          ;                       5166
 9372 032404 001171                        BNE     5$
 9373 032406 012701 125252                 MOV     #-52526,R1       ; *,TST.PAT             5169
 9374 032412 012702 000001                 MOV     #1,R2            ; *,TWICE               5171
 9375 032416 152777 000040 160570    1$:   BISB    #40,@ML.REG+40   ;                       5172
 9376 032424 016700 161152                 MOV     ML.DUT,R0
 9377 032430 042700 177770                 BIC     #177770,R0
 9378 032434 142777 000007 160552          BICB    #7,@ML.REG+40
 9379 032442 150077 160546                 BISB    R0,@ML.REG+40
 9380 032446 152777 000020 160540          BISB    #20,@ML.REG+40   ;                       5174
 9381 032454 010177 160524                 MOV     R1,@ML.REG+30    ; TST.PAT,*             5175
 9382 032460 132777 000010 160546          BITB    #10,@ML.REG+60   ;                       5177
 9383 032466 001022                        BNE     2$
 9384 032470 104455                        TRAP    55               ;                       5180
 9385 032472 000007                        .WORD   7
 9386 032474 010464                        .WORD   ASYNC
 9387 032476 024060                        .WORD   DUMPER
 9388 032500 012746 006374                 MOV     #WRD.9,-(SP)     ;                       5181
 9389 032504 012746 006344                 MOV     #WRD.7,-(SP)
 9390 032510 012746 006330                 MOV     #WRD.5,-(SP)
```

```
                                                                                                                    F 1
                                                                                                                    14-May-1981 13:48:54    TOPS
                                                                                                                    14-May-1981 13:05:08    PA:<
                                ;ML4
                                ;
                                                                                                                                            
032514  012746  006034                MOV     #THR.FMT,-(SP)
032520  012746  000004                MOV     #4,-(SP)
032524  010600                      MOV     SP,R0                                   ; SP,*
032526  104414                      TRAP    14
032530  062706  000012                ADD     #12,SP                                  ;                                               5179
032534  152777  000040  160452  2$:   BISB    #40,@ML.REG+40                          ;                                               5182
032542  016700  161034                MOV     ML.DUT,R0
032546  042700  177770                BIC     #177770,R0
032552  142777  000007  160434        BICB    #7,@ML.REG+40
032560  150077  160430                BISB    R0,@ML.REG+40
032564  010177  160414                MOV     R1,@ML.REG+30                           ; TST.PAT,*                                     5185
032570  132777  000010  160436        BITB    #10,@ML.REG+60                          ;                                               5187
032576  001422                      BEQ     3$
032600  104455                      TRAP    55                                      ;                                               5190
032602  000010                      .WORD   10
032604  010464                      .WORD   ASYNC
032606  024060                      .WORD   DUMPER
032610  012746  006374                MOV     #WRD.9,-(SP)                            ;                                               5191
032614  012746  006344                MOV     #WRD.7,-(SP)
032620  012746  006336                MOV     #WRD.6,-(SP)
032624  012746  006034                MOV     #THR.FMT,-(SP)
032630  012746  000004                MOV     #4,-(SP)
032634  010600                      MOV     SP,R0                                   ; SP,*
032636  104414                      TRAP    14
032640  062706  000012                ADD     #12,SP                                  ;                                               5189
032644  152777  000040  160342  3$:   BISB    #40,@ML.REG+40                          ;                                               5192
032652  016700  160724                MOV     ML.DUT,R0
032656  042700  177770                BIC     #177770,R0
032662  142777  000007  160324        BICB    #7,@ML.REG+40
032670  150077  160320                BISB    R0,@ML.REG+40
032674  017703  160304                MOV     @ML.REG+30,R3                           ; *,SAVE                                        5195
032700  032777  020000  160246        BIT     #20000,@ML.REG                          ;                                               5197
032706  001422                      BEQ     4$
032710  104455                      TRAP    55                                      ;                                               5200
032712  000011                      .WORD   11
032714  010464                      .WORD   ASYNC
032716  024060                      .WORD   DUMPER
032720  012746  006360                MOV     #WRD.8,-(SP)                            ;                                               5201
032724  012746  006344                MOV     #WRD.7,-(SP)
032730  012746  006336                MOV     #WRD.6,-(SP)
032734  012746  006034                MOV     #THR.FMT,-(SP)
032740  012746  000004                MOV     #4,-(SP)
032744  010600                      MOV     SP,R0                                   ; SP,*
032746  104414                      TRAP    14
032750  062706  000012                ADD     #12,SP                                  ;                                               5199
032754  006301                4$:   ASL     R1                                      ; TST.PAT                                       5204
032756  005202                      INC     R2                                      ; TWICE                                         5171
032760  020227  000002                CMP     R2,#2                                   ; TWICE,*
032764  003614                      BLE     1$
032766  000207                      RTS     PC                                      ;                                               5166
032770  012746  007074        5$:   MOV     #WRD.53,-(SP)                           ;                                               5209
032774  012746  006344                MOV     #WRD.7,-(SP)
```

```
                                ;ML4                                                    14-May-1981 13:48:54    TOPS
                                ;                                                       14-May-1981 13:05:08    PA:<
                                
033000  012746  006024                  MOV     #TWO.FMT,-(SP)
033004  012746  000003                  MOV     #3,-(SP)
033010  010600                          MOV     SP,R0                           ; SP,*
033012  104414                          TRAP    14
033014  062706  000010                  ADD     #10,SP
033020  000207                          RTS     PC                              ;                               5135

                                ; Routine Size:  139 words
                                ; Maximum stack depth per invocation:  9 words
                                
                                
                                
                                        .SBTTL  T10
033022                          T10::
033022  004767  177346          1$:     JSR     PC,$T10                         ;                               5209
033026  104466                          TRAP    66
033030  006000                          ROR     R0
033032  103773                          BLO     1$
033034  000207                          RTS     PC

                                ; Routine Size:  6 words
                                ; Maximum stack depth per invocation:  0 words
                                
                                
;       5212  !<BLF/PAGE>
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
:                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (61)

                5213  !
                5214  !
                5215  BGNTST;
                5216
                5217  !++
                5218  ! TEST NUMBER:  TST 11
                5219  !
                5220  ! TEST NAME:    MEMORY SIZING TEST
                5221  !
                5222  ! TEST DESCRIPTION:
                5223  !
                5224  !       THIS TESTS THE ML11'S SIZING
                5225  !       LOGIC BY COMPARING THE
                5226  !       OPERATORS INPUTED NUMBER OF ARRAYS
                5227  !       PRESENT TO THE ML11 SIZING
                5228  !       NUMBER OF ARRAYS PRESENT
                5229  !       THE DRIVE IS DROPPED ON DETECTED ERRORS.
                5230  ! IMPLICIT INPUTS:
                5231  !
                5232  !       OP_NUM_ARR:
                5233  !       LOADED DURING INIT CODE AND
                5234  !       STORES OPERATORS INPUT TO THE
                5235  !       SOFTWARE QUESTION:
                5236  !       NUMBER OF ARRAYS PRESENT?
                5237  !
                5238  !
                5239  !--
                5240
                5241  local
                5242      DODU_FLG;                                   !DROP UNIT FLAG
                5243
                5244  BGNSUB;
                5245  CLR_MBUS;
                5246  DODU_FLG = ZERO;
                5247
                5248  if (.OP_NUM_ARR + 1) neq .ML_NUM_ARR            !SEE IF DRIVE SIZED SAME NO. OF ARRAYS AS OP INPUTED
                5249  then
                5250      begin
                5251      ERRDF (10, ASYNC, DUMPER);                  !IF NOT EQL THEN ERROR AND SET DODU_FLG
                5252      PRINTB (TWO_FMT, FNC_1, WRD_14);
                5253      PRINTB (FMT_2, (.OP_NUM_ARR + 1), .ML_NUM_ARR);
                5254      DODU_FLG = ONE;
                5255      end;
                5256
                5257  ENDSUB;
                5258
                5259  if .DODU_FLG IS_SET                           !DROP THIS UNIT IF DODU_FLG IS_SET
                5260  then
                5261      begin
                5262      DODU (.ML_LUN);
                5263      DOCLN;
                5264      end;
```

```
                                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
                                                                    14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (61)

9547 ;ML4
9548 ;
9549
9550 ;          5265
9551 ;          5266  ENDTST;
9555
9556                                        .SBTTL  $T11
9560 033036  004167  151016          $T11:  JSR     R1,$SAVE3                    ;                                  5211
9561 033042  104402                  1$:    TRAP    2                            ;                                  5242
9562 033044  152777  000040  160142         BISB    #40,@ML.REG+40               ;                                  5244
9563 033052  016701  160524                 MOV     ML.DUT,R1
9564 033056  042701  177770                 BIC     #177770,R1
9565 033062  142777  000007  160124         BICB    #7,@ML.REG+40
9566 033070  150177  160120                 BISB    R1,@ML.REG+40
9567 033074  005002                         CLR     R2                           ; DODU.FLG                         5246
9568 033076  016701  156456                 MOV     OP.NUM.ARR,R1                ;                                  5248
9569 033102  005201                         INC     R1
9570 033104  017703  160164                 MOV     @ML.REG+120,R3
9571 033110  006203                         ASR     R3
9572 033112  006203                         ASR     R3
9573 033114  006203                         ASR     R3
9574 033116  000303                         SWAB    R3
9575 033120  042703  177740                 BIC     #177740,R3
9576 033124  010100                         MOV     R1,R0
9577 033126  020003                         CMP     R0,R3
9578 033130  001444                         BEQ     2$
9579 033132  104455                         TRAP    55                           ;                                  5251
9580 033134  000012                         .WORD   12
9581 033136  010464                         .WORD   ASYNC
9582 033140  024060                         .WORD   DUMPER
9583 033142  012746  006442                 MOV     #WRD.14,-(SP)                ;                                  5252
9584 033146  012746  007726                 MOV     #FNC.1,-(SP)
9585 033152  012746  006024                 MOV     #TWO.FMT,-(SP)
9586 033156  012746  000003                 MOV     #3,-(SP)
9587 033162  010600                         MOV     SP,R0                        ; SP,*
9588 033164  104414                         TRAP    14
9589 033166  017703  160102                 MOV     @ML.REG+120,R3               ;                                  5253
9590 033172  006203                         ASR     R3
9591 033174  006203                         ASR     R3
9592 033176  006203                         ASR     R3
9593 033200  000303                         SWAB    R3
9594 033202  042703  177740                 BIC     #177740,R3
9595 033206  010316                         MOV     R3,(SP)
9596 033210  016746  156344                 MOV     OP.NUM.ARR,-(SP)
9597 033214  005216                         INC     (SP)
9598 033216  012746  004266                 MOV     #FMT.2,-(SP)
9599 033222  012746  000003                 MOV     #3,-(SP)
9600 033226  010600                         MOV     SP,R0                        ; SP,*
9601 033230  104414                         TRAP    14
```

```
 9603                                   ;ML4                                                                    14-May-1981 13:48:54    TOPS
 9604                                   ;                                                                       14-May-1981 13:05:08    PA:<
 9605
 9606 033232  012702  000001                    MOV     #1,R2                          ; *,DODU.FLG                                    5254
 9607 033236  062706  000016                    ADD     #16,SP                         ;                                               5250
 9608 033242  104467                    2$:     TRAP    67                             ;                                               5255
 9609 033244  006000                            ROR     R0
 9610 033246  103675                            BLO     1$
 9611 033250  005302                            DEC     R2                             ; DODU.FLG                                      5259
 9612 033252  001004                            BNE     3$
 9613 033254  016700  160320                    MOV     ML.LUN,R0                      ;                                               5262
 9614 033260  104451                            TRAP    51
 9615 033262  104444                            TRAP    44
 9616 033264  000207                    3$:     RTS     PC                             ;                                               5211
 9617
 9618                                   ; Routine Size:  76 words
 9619                                   ; Maximum stack depth per invocation:  11 words
 9624
 9625
 9629
 9630                                           .SBTTL  T11
 9634 033266                            T11::
 9635 033266  004767  177544            1$:     JSR     PC,$T11                        ;                                               5264
 9636 033272  104466                            TRAP    66
 9637 033274  006000                            ROR     R0
 9638 033276  103773                            BLO     1$
 9639 033300  000207                            RTS     PC
 9640
 9641                                   ; Routine Size:  6 words
 9642                                   ; Maximum stack depth per invocation:  0 words
 9647
 9648
 9649 ;          5267   !<BLF/PAGE>
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (62)

```
;ML4
;
;                5268  !
;                5269  !
;                5270  BGNTST;
;                5271
;                5272  !++
;                5273  ! TEST NUMBER:  TST 12
;                5274  !
;                5275  ! TEST NAME:    NO-OP FUNCTION TEST
;                5276  !
;                5277  ! TEST DESCRIPTION:
;                5278  !
;                5279  !       TEST IF THE DRIVE CAN PERFORM
;                5280  !       A NO_OP FUNCTION WITH OUT
;                5281  !       HANGING THE DRIVE.
;                5282  !
;                5283  !       A NO_OP FUNCTION IS WRITTEN
;                5284  !       INTO MLCS1.
;                5285  !       THEN GO AND ERROR BITS ARE
;                5286  !       CHECKED FOR CORRECT STATES.
;                5287  !
;                5288  !       THIS UNIT IS DROPPED ON DETECTED
;                5289  !       ERRORS.
;                5290  !
;                5291  ! IMPLICIT INPUTS:      NONE
;                5292  !
;                5293  !
;                5294  !--
;                5295
;                5296  local
;                5297      DODU_FLG;                                   !DROP UNIT FLAG
;                5298
;                5299  BGNSUB;
;                5300  CLR_MBUS;
;                5301  DODU_FLG = ZERO;
;                5302  MLCST = NOOP;                                   !DO A NOOP FUNCTION
;                5303  DELAY (ONE_US);                                 !DELAY 1 US
;                5304
;                5305  if .GO IS_SET                                   !SEE IF GO STILL SET
;                5306  then
;                5307      begin
;                5308      ERRDF (11, ASYNC, DUMPER);                  !ERROR AND SET DODU_FLG IF STILL SET
;                5309      PRINTB (FIV_FMT, WRD_1, PHR_2, WRD_11, FNC_2, WRD_19);
;                5310      DODU_FLG = ONE;
;                5311      end;
;                5312
;                5313  if .ILF IS_SET                                  !SEE ILF SET
;                5314  then
;                5315      begin
;                5316      ERRDF (12, ASYNC, DUMPER);                  !ERROR AND SET DODU_FLG IF SET
;                5317      PRINTB (FIV_FMT, WRD_3, PHR_5, WRD_11, FNC_2, WRD_19);
;                5318      end;
;                5319
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (62)

;          5320  if .OPI IS_SET                        !SEE IF CPI SET
;          5321  then
;          5322      begin
;          5323      ERRDF (13, ASYNC, DUMPER);        !ERROR AND SET DODU_FLG IF SET
;          5324      PRINTB (FIV_FMT, WRD_4, PHR_5, WRD_11, FNC_2, WRD_19);
;          5325      end;
;          5326
;          5327  ENDSUB;
;          5328
;          5329  if .DODU_FLG IS_SET                   !DROP THIS UNIT IF DODU_FLG IS_SET
;          5330  then
;          5331      begin
;          5332      DODU (.ML_LUN);
;          5333      DOCLN;
;          5334      end;
;          5335
;          5336  ENDTST;

                                    .SBTTL  $T12
033302  004167  150536        $T12: JSR     R1,$SAVE2                 ;                    5266
033306  005746                    TST     -(SP)
033310  104402                1$: TRAP    2                         ;                    5297
033312  152777  000040  157674    BISB    #40,@ML.REG+40            ;                    5299
033320  016701  160256            MOV     ML.DUT,R1
033324  042701  177770            BIC     #177770,R1
033330  142777  000007  157656    BICB    #7,@ML.REG+40
033336  150177  157652            BISB    R1,@ML.REG+40
033342  005002                    CLR     R2                        ; DODU.FLG           5301
033344  012777  000001  157602    MOV     #1,@ML.REG                ;                    5302
033352  012700  000001            MOV     #1,R0                     ; *,$$TMP2           5303
033356  001410              2$:   BEQ     5$
033360  016701  146532            MOV     L$DLY,R1                  ; *,$$TMP1
033364  001403                    BEQ     4$
033366  005016              3$:   CLR     (SP)                      ; $$TMP
033370  005301                    DEC     R1                        ; $$TMP1
033372  001375                    BNE     3$
033374  005300              4$:   DEC     R0                        ; $$TMP2
033376  000767                    BR      2$
033400  132777  000001  157546  5$: BITB    #1,@ML.REG                ;                    5305
033406  001430                    BEQ     6$
033410  104455                    TRAP    55                        ;                    5308
033412  000013                    .WORD   13
033414  010464                    .WORD   ASYNC
033416  024060                    .WORD   DUMPER
033420  012746  006506            MOV     #WRD.19,-(SP)             ;                    5309
033424  012746  007742            MOV     #FNC.2,-(SP)
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                                MOV     #WRD.11,-(SP)
033430  012746  006416                        MOV     #PHR.2,-(SP)
033434  012746  007404                        MOV     #WRD.1,-(SP)
033440  012746  006276                        MOV     #FIV.FMT,-(SP)
033444  012746  006062                        MOV     #6,-(SP)
033450  012746  000006                        MOV     SP,R0                           ; SP,*
033454  010600                                TRAP    14
033456  104414                                MOV     #1,R2                           ; *,DODU.FLG                            5310
033460  012702  000001                        ADD     #16,SP                          ;                                       5307
033464  062706  000016                        BITB    #1,@ML.REG+60                   ;                                       5313
033470  132777  000001  157536  6$:           BEQ     7$
033476  001426                                TRAP    55                              ;                                       5316
033500  104455                                .WORD   14
033502  000014                                .WORD   ASYNC
033504  010464                                .WORD   DUMPER
033506  024060                                MOV     #WRD.19,-(SP)                   ;                                       5317
033510  012746  006506                        MOV     #FNC.2,-(SP)
033514  012746  007742                        MOV     #WRD.11,-(SP)
033520  012746  006416                        MOV     #PHR.5,-(SP)
033524  012746  007472                        MOV     #WRD.3,-(SP)
033530  012746  006314                        MOV     #FIV.FMT,-(SP)
033534  012746  006062                        MOV     #6,-(SP)
033540  012746  000006                        MOV     SP,R0                           ; SP,*
033544  010600                                TRAP    14
033546  104414                                ADD     #16,SP                          ;                                       5315
033550  062706  000016                        BIT     #20000,@ML.REG+60               ;                                       5320
033554  032777  020000  157452  7$:           BEQ     8$
033562  001426                                TRAP    55                              ;                                       5323
033564  104455                                .WORD   15
033566  000015                                .WORD   ASYNC
033570  010464                                .WORD   DUMPER
033572  024060                                MOV     #WRD.19,-(SP)                   ;                                       5324
033574  012746  006506                        MOV     #FNC.2,-(SP)
033600  012746  007742                        MOV     #WRD.11,-(SP)
033604  012746  006416                        MOV     #PHR.5,-(SP)
033610  012746  007472                        MOV     #WRD.4,-(SP)
033614  012746  006322                        MOV     #FIV.FMT,-(SP)
033620  012746  006062                        MOV     #6,-(SP)
033624  012746  000006                        MOV     SP,R0                           ; SP,*
033630  010600                                TRAP    14
033632  104414                                ADD     #16,SP                          ;                                       5322
033634  062706  000016                        TRAP    67                              ;                                       5325
033640  104467                        8$:     ROR     R0
033642  006000                                BLO     1$
033644  103621                                DEC     R2                              ; DODU.FLG                              5329
033646  005302                                BNE     9$
033650  001004                                MOV     ML.LUN,R0                       ;                                       5332
033652  016700  157722                        TRAP    51
033656  104451                                TRAP    44
033660  104444                        9$:     TST     (SP)+                           ;                                       5266
033662  005726                                RTS     PC
033664  000207
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<

                                        ; Routine Size:  122 words
                                        ; Maximum stack depth per invocation:  11 words


                                                .SBTTL  T12
033666                                  T12::
033666  004767  177410                  1$:     JSR     PC,$T12                         ;       5334
033672  104466                                  TRAP    66
033674  006000                                  ROR     R0
033676  103773                                  BLO     1$
033700  000207                                  RTS     PC

                                        ; Routine Size:  6 words
                                        ; Maximum stack depth per invocation:  0 words


;       5337  !<BLF/PAGE>
```

```
 9855 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
 9856 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (63)
 9857
 9858 ;          5338  !
 9859 ;          5339  !
 9860 ;          5340  BGNTST;
 9861 ;          5341
 9862 ;          5342  !++
 9863 ;          5343  ! TEST NUMBER:  TST 13
 9864 ;          5344  !
 9865 ;          5345  ! TEST NAME:    WRITE CHECK FUNCTION TEST
 9866 ;          5346  !
 9867 ;          5347  ! TEST DESCRIPTION:
 9868 ;          5348  !
 9869 ;          5349  !       TEST IF THE DRIVE CAN PERFORM
 9870 ;          5350  !       A WRITE CHECK FUNCTION WITHOUT
 9871 ;          5351  !       HANGING THE DRIVE.
 9872 ;          5352  !
 9873 ;          5353  !       A WRITE CHECK FUNCTION IS WRITTEN
 9874 ;          5354  !       INTO MLCS1.
 9875 ;          5355  !       THEN GO AND ERROR BITS ARE
 9876 ;          5356  !       CHECKED FOR CORRECT STATES
 9877 ;          5357  !
 9878 ;          5358  !--
 9879 ;          5359
 9880 ;          5360  local
 9881 ;          5361      DODU_FLG;                                  !DROP UNIT FLAG
 9882 ;          5362
 9883 ;          5363  BGNSUB;
 9884 ;          5364  CLR_MBUS;
 9885 ;          5365  DODU_FLG = ZERO;
 9886 ;          5366  FIRST_BLK_XFER ();                           !SET UP A FIRST BLK XFERR
 9887 ;          5367  MLCS1 = WRT_CHK;                             !DO A WRITE CHECK FUNCTION
 9888 ;          5368
 9889 ;          5369  if .GO IS_NOT_SET                          !SEE IF THE GO BIT GOT SET
 9890 ;          5370  then
 9891 ;          5371      begin
 9892 ;          5372      ERRDF (14, ASYNC, DUMPER);             !ERROR IF NOT SET
 9893 ;          5373      PRINTB (FIV_FMT, WRD_1, PHR_1, WRD_12, FNC_4, WRD_19);
 9894 ;          5374
 9895 ;          5375      if .DRY IS_NOT_SET                     !SEE IF THE DRY IS SET WHILE GO IS CLEARED
 9896 ;          5376      then
 9897 ;          5377          begin
 9898 ;          5378          ERRDF (15, ASYNC, DUMPER);         !ERROR IF CLEARED
 9899 ;          5379          PRINTB (FIV_FMT, WRD_2, PHR_1, WRD_43, WRD_1, PHR_6);
 9900 ;          5380          end;
 9901 ;          5381
 9902 ;          5382      DODU_FLG = ONE;                        !SET DODU_FLG
 9903 ;          5383      end
 9904 ;          5384  else
 9905 ;          5385
 9906 ;          5386      if .DRY IS_SET                         !THE GO IS SET SO SEE IF DRY IS CLEARED
 9907 ;          5387      then
 9908 ;          5388          begin
 9909 ;          5389          ERRDF (16, ASYNC, DUMPER);         !ERROR IF DRY IS SET
```

```
9911 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
9912 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (63)
9913
9914 ;          5390          PRINTB (FIV_FMT, WRD_2, PHR_2, WRD_43, WRD_1, PHR_5);
9915 ;          5391          DODU_FLG = ONE;
9916 ;          5392          end;
9917 ;          5393
9918 ;          5394  if .ILF IS_SET                                 !SEE IF ILF IS SET
9919 ;          5395  then
9920 ;          5396      begin
9921 ;          5397      ERRDF (17, ASYNC, DUMPER);                 !ERROR IF SET
9922 ;          5398      PRINTB (FIV_FMT, WRD_3, PHR_5, WRD_12, FNC_4, WRD_19);
9923 ;          5399      end;
9924 ;          5400
9925 ;          5401  if .OPI IS_SET                                 !SEE IF OPI IS SET
9926 ;          5402  then
9927 ;          5403      begin
9928 ;          5404      ERRDF (18, ASYNC, DUMPER);                 !ERROR IF SET
9929 ;          5405      PRINTB (FIV_FMT, WRD_4, PHR_5, WRD_12, FNC_4, WRD_19);
9930 ;          5406      end;
9931 ;          5407
9932 ;          5408  DELAY (FRTY_US);                               !WAIT FO XFERR TO COMPLTE
9933 ;          5409
9934 ;          5410  if .DRY IS_NOT_SET                             !SEE IF DRY IS SETS AFTER XFERR
9935 ;          5411  then
9936 ;          5412      begin
9937 ;          5413
9938 ;          5414      if .GO IS_SET                              !IF DRY IS NOT SET THEN SEE IF GO IS SET
9939 ;          5415      then
9940 ;          5416          begin
9941 ;          5417          CLR_MBUS;                              !IF THE GO IS SET THEN TRY TO CLR GO
9942 ;          5418
9943 ;          5419          if .GO IS_SET then ERRDF (19, ASYNC, DUMPER) else ERRDF (20, SYNC, DUMPER);
9944 ;          5420
9945 ;          5421                                                 !IF GO IS STILL SET THEN ASYNC FAILURE
9946 ;          5422          PRINTB (FIV_FMT, WRD_1, PHR_2, WRD_11, FNC_4, WRD_19);  !ELSE SYNC MODLE FAILURE
9947 ;          5423          end
9948 ;          5424      else
9949 ;          5425          begin
9950 ;          5426          ERRDF (21, ASYNC, DUMPER);             !ERROR GO AND DRY BOTH CLEARED
9951 ;          5427          PRINTB (FIV_FMT, WRD_2, PHR_1, WRD_11, FNC_4, WRD_19);
9952 ;          5428          end;
9953 ;          5429
9954 ;          5430      DODU_FLG = ONE;                            !SET THE DODU_FLG
9955 ;          5431      end;
9956 ;          5432
9957 ;          5433  if .GO IS_SET                                 !SEE IF THE GO IS STILL SET
9958 ;          5434  then
9959 ;          5435      begin
9960 ;          5436      CLR_MBUS;                                  !TRY TO CLR GO IF STILL SET
9961 ;          5437
9962 ;          5438      if .GO IS_SET then ERRDF (22, ASYNC, DUMPER) else ERRDF (23, SYNC, DUMPER);
9963 ;          5439
9964 ;          5440                                                 !IF GO IS STILL SET THE ASYNC FAILURE
9965 ;          5441      PRINTB (FIV_FMT, WRD_1, PHR_2, WRD_11, FNC_4, WRD_19);      !ELSE SYNC MODULE FAILURE
```

```
 9967 ;ML4                                                          14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
 9968 :                                                             14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (63)
 9969
 9970 :          5442      DODU_FLG = ONE;
 9971 :          5443      end;
 9972 :          5444
 9973 :          5445  ENDSUB;
 9974 :          5446
 9975 :          5447  if .DODU_FLG IS_SET                          !DROP THIS UNIT IF DODU_FLG IS_SET
 9976 :          5448  then
 9977 :          5449      begin
 9978 :          5450      DODU (.ML_LUN);
 9979 :          5451      DOCLN;
 9980 :          5452      end;
 9981 :          5453
 9982 :          5454  ENDTST;
 9986
 9987                                          .SBTTL  $T13
 9991 033702  004167  150136          $T13:   JSR     R1,$SAVE2                       ;                               5336
 9992 033706  005746                          TST     -(SP)
 9993 033710  104402                  1$:     TRAP    2                               ;                               5361
 9994 033712  152777  000040  157274          BISB    #40,@ML.REG+40                  ;                               5363
 9995 033720  016701  157656                  MOV     ML.DUT,R1
 9996 033724  042701  177770                  BIC     #177770,R1
 9997 033730  142777  000007  157256          BICB    #7,@ML.REG+40
 9998 033736  150177  157252                  BISB    R1,@ML.REG+40
 9999 033742  005002                          CLR     R2                              ; DODU.FLG                      5365
10000 033744  004767  162070                  JSR     PC,FIRST.BLK.XFER               ;                               5366
10001 033750  012777  000051  157176          MOV     #51,@ML.REG                     ;                               5367
10002 033756  132777  000001  157170          BITB    #1,@ML.REG                      ;                               5369
10003 033764  001057                          BNE     2$
10004 033766  104455                          TRAP    55                              ;                               5372
10005 033770  000016                          .WORD   16
10006 033772  010464                          .WORD   ASYNC
10007 033774  024060                          .WORD   DUMPER
10008 033776  012746  006506                  MOV     #WRD.19,-(SP)                   ;                               5373
10009 034002  012746  007756                  MOV     #FNC.4,-(SP)
10010 034006  012746  006426                  MOV     #WRD.12,-(SP)
10011 034012  012746  007366                  MOV     #PHR.1,-(SP)
10012 034016  012746  006276                  MOV     #WRD.1,-(SP)
10013 034022  012746  006062                  MOV     #FIV.FMT,-(SP)
10014 034026  012746  000006                  MOV     #6,-(SP)
10015 034032  010600                          MOV     SP,R0                           ; SP,*
10016 034034  104414                          TRAP    14
10017 034036  132777  000200  157160          BITB    #200,@ML.REG+50                 ;                               5375
10018 034044  001056                          BNE     3$
10019 034046  104455                          TRAP    55                              ;                               5378
10020 034050  000017                          .WORD   17
10021 034052  010464                          .WORD   ASYNC
```

```
10023                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
10024                                   ;                                                       14-May-1981 13:05:08    PA:<
10025
10026 034054  024060                            .WORD   DUMPER
10027 034056  012746  007504                      MOV     #PHR.6,-(SP)          ;                                 5379
10028 034062  012746  006276                      MOV     #WRD.1,-(SP)
10029 034066  012746  006754                      MOV     #WRD.43,-(SP)
10030 034072  012746  007366                      MOV     #PHR.1,-(SP)
10031 034076  012746  006302                      MOV     #WRD.2,-(SP)
10032 034102  012746  006062                      MOV     #FIV.FMT,-(SP)
10033 034106  012746  000006                      MOV     #6,-(SP)
10034 034112  010600                            MOV     SP,R0                 ; SP,*
10035 034114  104414                            TRAP    14
10036 034116  062706  000016                      ADD     #16,SP                ;                                 5377
10037 034122  000427                            BR      3$                    ;                                 5382
10038 034124  105777  157074          2$:       TSTB    @ML.REG+50            ;                                 5386
10039 034130  100030                            BPL     4$
10040 034132  104455                            TRAP    55                    ;                                 5389
10041 034134  000020                            .WORD   20
10042 034136  010464                            .WORD   ASYNC
10043 034140  024060                            .WORD   DUMPER
10044 034142  012746  007472                      MOV     #PHR.5,-(SP)          ;                                 5390
10045 034146  012746  006276                      MOV     #WRD.1,-(SP)
10046 034152  012746  006754                      MOV     #WRD.43,-(SP)
10047 034156  012746  007404                      MOV     #PHR.2,-(SP)
10048 034162  012746  006302                      MOV     #WRD.2,-(SP)
10049 034166  012746  006062                      MOV     #FIV.FMT,-(SP)
10050 034172  012746  000006                      MOV     #6,-(SP)
10051 034176  010600                            MOV     SP,R0                 ; SP,*
10052 034200  104414                            TRAP    14
10053 034202  012702  000001          3$:       MOV     #1,R2                 ; *,DODU.FLG                      5391
10054 034206  062706  000016                      ADD     #16,SP                ;                                 5388
10055 034212  132777  000001  157014  4$:       BITB    #1,@ML.REG+60         ;                                 5394
10056 034220  001426                            BEQ     5$
10057 034222  104455                            TRAP    55                    ;                                 5397
10058 034224  000021                            .WORD   21
10059 034226  010464                            .WORD   ASYNC
10060 034230  024060                            .WORD   DUMPER
10061 034232  012746  006506                      MOV     #WRD.19,-(SP)         ;                                 5398
10062 034236  012746  007756                      MOV     #FNC.4,-(SP)
10063 034242  012746  006426                      MOV     #WRD.12,-(SP)
10064 034246  012746  007472                      MOV     #PHR.5,-(SP)
10065 034252  012746  006314                      MOV     #WRD.3,-(SP)
10066 034256  012746  006062                      MOV     #FIV.FMT,-(SP)
10067 034262  012746  000006                      MOV     #6,-(SP)
10068 034266  010600                            MOV     SP,R0                 ; SP,*
10069 034270  104414                            TRAP    14
10070 034272  062706  000016                      ADD     #16,SP                ;                                 5396
10071 034276  032777  020000  156730  5$:       BIT     #20000,@ML.REG+60     ;                                 5401
10072 034304  001426                            BEQ     6$
10073 034306  104455                            TRAP    55                    ;                                 5404
10074 034310  000022                            .WORD   22
10075 034312  010464                            .WORD   ASYNC
10076 034314  024060                            .WORD   DUMPER
10077 034316  012746  006506                      MOV     #WRD.19,-(SP)         ;                                 5405
```

```
10079                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
10080                                   ;                                                       14-May-1981 13:05:08    PA:<
10081
10082 034322  012746  007756                     MOV     #FNC.4,-(SP)
10083 034326  012746  006426                     MOV     #WRD.12,-(SP)
10084 034332  012746  007472                     MOV     #PHR.5,-(SP)
10085 034336  012746  006322                     MOV     #WRD.4,-(SP)
10086 034342  012746  006062                     MOV     #FIV.FMT,-(SP)
10087 034346  012746  000006                     MOV     #6,-(SP)
10088 034352  010600                           MOV     SP,R0                   ; SP,*
10089 034354  104414                           TRAP    14
10090 034356  062706  000016                     ADD     #16,SP                  ;                                       5403
10091 034362  012700  000050             6$:     MOV     #50,R0                  ; *,$$TMP2                              5408
10092 034366  001410                     7$:     BEQ     10$
10093 034370  016701  145522                     MOV     L$DLY,R1                ; *,$$TMP1
10094 034374  001403                           BEQ     9$
10095 034376  005016                     8$:     CLR     (SP)                    ; $$TMP
10096 034400  005301                           DEC     R1                      ; $$TMP1
10097 034402  001375                           BNE     8$
10098 034404  005300                     9$:     DEC     R0                      ; $$TMP2
10099 034406  000767                           BR      7$
10100 034410  132777  000200  156606   10$:    BITB    #200,@ML.REG+50         ;                                       5410
10101 034416  001106                           BNE     15$
10102 034420  132777  000001  156526           BITB    #1,@ML.REG              ;                                       5414
10103 034426  001452                           BEQ     13$
10104 034430  152777  000040  156556           BISB    #40,@ML.REG+40          ;                                       5416
10105 034436  016701  157140                     MOV     ML.DUT,R1
10106 034442  042701  177770                     BIC     #177770,R1
10107 034446  142777  000007  156540           BICB    #7,@ML.REG+40
10108 034454  150177  156534                     BISB    R1,@ML.REG+40
10109 034460  132777  000001  156466           BITB    #1,@ML.REG              ;                                       5419
10110 034466  001405                           BEQ     11$
10111 034470  104455                           TRAP    55
10112 034472  000023                           .WORD   23
10113 034474  010464                           .WORD   ASYNC
10114 034476  024060                           .WORD   DUMPER
10115 034500  000404                           BR      12$
10116 034502  104455                    11$:    TRAP    55
10117 034504  000024                           .WORD   24
10118 034506  010526                           .WORD   SYNC
10119 034510  024060                           .WORD   DUMPER
10120 034512  012746  006506             12$:    MOV     #WRD.19,-(SP)           ;                                       5422
10121 034516  012746  007756                     MOV     #FNC.4,-(SP)
10122 034522  012746  006416                     MOV     #WRD.11,-(SP)
10123 034526  012746  007404                     MOV     #PHR.2,-(SP)
10124 034532  012746  006276                     MOV     #WRD.1,-(SP)
10125 034536  012746  006062                     MOV     #FIV.FMT,-(SP)
10126 034542  012746  000006                     MOV     #6,-(SP)
10127 034546  010600                           MOV     SP,R0                   ; SP,*
10128 034550  104414                           TRAP    14
10129 034552  000424                           BR      14$                     ;                                       5414
10130 034554  104455                    15$:    TRAP    55                      ;                                       5426
10131 034556  000025                           .WORD   25
10132 034560  010464                           .WORD   ASYNC
10133 034562  024060                           .WORD   DUMPER
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                                                                                                        5427
034564  012746  006506                        MOV     #WRD.19,-(SP)           ;
034570  012746  007756                        MOV     #FNC.4,-(SP)
034574  012746  006416                        MOV     #WRD.11,-(SP)
034600  012746  007366                        MOV     #PHR.1,-(SP)
034604  012746  006302                        MOV     #WRD.2,-(SP)
034610  012746  006062                        MOV     #FIV.FMT,-(SP)
034614  012746  000006                        MOV     #6,-(SP)
034620  010600                                MOV     SP,R0                   ; SP,*
034622  104414                                TRAP    14
034624  012702  000001                14$:    MOV     #1,R2                   ; *,DODU.FLG                    5430
034630  062706  000016                        ADD     #16,SP                  ;                               5412
034634  132777  000001  156312        15$:    BITB    #1,@ML.REG              ;                               5433
034642  001455                                BEQ     18$
034644  152777  000040  156342                BISB    #40,@ML.REG+40          ;                               5435
034652  016701  156724                        MOV     ML.DUT,R1
034656  042701  177770                        BIC     #177770,R1
034662  142777  000007  156324                BICB    #7,@ML.REG+40
034670  150177  156320                        BISB    R1,@ML.REG+40
034674  132777  000001  156252                BITB    #1,@ML.REG              ;                               5438
034702  001405                                BEQ     16$
034704  104455                                TRAP    55
034706  000026                                .WORD   26
034710  010464                                .WORD   ASYNC
034712  024060                                .WORD   DUMPER
034714  000404                                BR      17$
034716  104455                        16$:    TRAP    55
034720  000027                                .WORD   27
034722  010526                                .WORD   SYNC
034724  024060                                .WORD   DUMPER
034726  012746  006506                17$:    MOV     #WRD.19,-(SP)           ;                               5441
034732  012746  007756                        MOV     #FNC.4,-(SP)
034736  012746  006416                        MOV     #WRD.11,-(SP)
034742  012746  007404                        MOV     #PHR.2,-(SP)
034746  012746  006276                        MOV     #WRD.1,-(SP)
034752  012746  006062                        MOV     #FIV.FMT,-(SP)
034756  012746  000006                        MOV     #6,-(SP)
034762  010600                                MOV     SP,R0                   ; SP,*
034764  104414                                TRAP    14
034766  012702  000001                        MOV     #1,R2                   ; *,DODU.FLG                    5442
034772  062706  000016                        ADD     #16,SP                  ;                               5435
034776  104467                        18$:    TRAP    67                      ;                               5443
035000  006000                                ROR     R0
035002  103002                                BHIS    19$
035004  000167  176700                        JMP     1$
035010  005302                        19$:    DEC     R2                      ; DODU.FLG                      5447
035012  001004                                BNE     20$
035014  016700  156560                        MOV     ML.LUN,R0               ;                               5450
035020  104451                                TRAP    51
035022  104444                                TRAP    44
035024  005726                        20$:    TST     (SP)+                   ;                               5336
035026  000207                                RTS     PC
```

```
10191                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
10192                                   ;                                                       14-May-1981 13:05:08    PA:<
10193
10194                                   ; Routine Size:  299 words
10195                                   ; Maximum stack depth per invocation:  18 words
10200
10201
10205
10206                                           .SBTTL  T13
10210 035030                            T13::
10211 035030  004767  176646            1$:     JSR     PC,$T13                         ;       5452
10212 035034  104466                            TRAP    66
10213 035036  006000                            ROR     R0
10214 035040  103773                            BLO     1$
10215 035042  000207                            RTS     PC
10216
10217                                   ; Routine Size:  6 words
10218                                   ; Maximum stack depth per invocation:  0 words
10223
10224
10225 ;         5455  !<BLF/PAGE>
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (64)

```
10227 ;ML4
10228 ;
10229
10230 ;          5456  !
10231 ;          5457  BGNTST;
10232 ;          5458
10233 ;          5459  !++
10234 ;          5460  ! TEST NUMBER:  TST 14
10235 ;          5461  !
10236 ;          5462  ! TEST NAME:    WRITE FUNCTION TEST
10237 ;          5463  !
10238 ;          5464  ! TEST DESCRIPTION:
10239 ;          5465  !       TEST IF THE DRIVE CAN PERFORM A WRITE FUNCTION WITHOUT
10240 ;          5466  !       HANGING THE DRIVE.
10241 ;          5467  !
10242 ;          5468  !       A WRITE FUNCTION IS WRITTEN INTO MLCS1.  THEN GO AND ERROR BITS ARE
10243 ;          5469  !       CHECKED FOR CORRECT STATUS.  THIS UNIT IS DROPPED ON DETECTED ERRORS.
10244 ;          5470  !--
10245 ;          5471
10246 ;          5472  local
10247 ;          5473      DODU_FLG;                                   !DROP UNIT FLAG
10248 ;          5474
10249 ;          5475  BGNSUB;
10250 ;          5476  CLR_MBUS;
10251 ;          5477  DODU_FLG = ZERO;
10252 ;          5478  FIRST_BLK_XFER ();                              !SET UP A FIRST BLOCK XFERR
10253 ;          5479  MLCS1 = write;                                  !DO A WRITE FUNCTION
10254 ;          5480
10255 ;          5481  if .GO IS_NOT_SET                               !SEE IF THE GO IS SET
10256 ;          5482  then
10257 ;          5483      begin                                       !ERROR IF NOT SET
10258 ;          5484      ERRDF (24, ASYNC, DUMPER);
10259 ;          5485      PRINTB (FIV_FMT, WRD_1, PHR_1, WRD_12, FNC_5, WRD_19);
10260 ;          5486
10261 ;          5487      if .DRY IS_NOT_SET                          !SEE IF DRY SET WITH GO CLEAR
10262 ;          5488      then
10263 ;          5489          begin
10264 ;          5490          ERRDF (25, ASYNC, DUMPER);
10265 ;          5491          PRINTB (FIV_FMT, WRD_2, PHR_1, WRD_43, WRD_1, PHR_6);
10266 ;          5492          end;
10267 ;          5493
10268 ;          5494      DODU_FLG = ONE;                             !SET DODU_FLG
10269 ;          5495      end
10270 ;          5496  else                                            !GO IS SET DURING FUNCTION
10271 ;          5497
10272 ;          5498      if .DRY IS_SET                              !SEE IF DRY CLEAR WITH GO SET
10273 ;          5499      then
10274 ;          5500          begin                                   !ERROR IF SET
10275 ;          5501          ERRDF (26, ASYNC, DUMPER);
10276 ;          5502          PRINTB (FIV_FMT, WRD_2, PHR_2, WRD_43, WRD_1, PHR_5);
10277 ;          5503          DODU_FLG = ONE;
10278 ;          5504          end;
10279 ;          5505
10280 ;          5506  if .ILF IS_SET                                 !DID FUNCTION CAUSE ILF
10281 ;          5507  then
```

```
10283 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
10284 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (64)
10285
10286 ;          5508      begin                              !ERROR IF YES
10287 ;          5509      ERRDF (27, ASYNC, DUMPER);
10288 ;          5510      PRINTB (FIV_FMT, WRD_3, PHR_5, WRD_12, FNC_5, WRD_19);
10289 ;          5511      end;
10290 ;          5512
10291 ;          5513  if .OPI IS_SET                          !DID FUNCTION CAUSE OPI
10292 ;          5514  then
10293 ;          5515      begin                              !ERROR IF YES
10294 ;          5516      ERRDF (28, ASYNC, DUMPER);
10295 ;          5517      PRINTB (FIV_FMT, WRD_4, PHR_5, WRD_12, FNC_5, WRD_19);
10296 ;          5518      end;
10297 ;          5519
10298 ;          5520  DELAY (FRTY_US);                       !WAIT FOR XFERR TC COMPLETE
10299 ;          5521
10300 ;          5522  if .DRY IS_NOT_SET                     !SEE IF DRY CLEARED AFTER XFERR
10301 ;          5523  then
10302 ;          5524      begin
10303 ;          5525
10304 ;          5526      if .GO IS_SET                      !TST GO CLR IF DRY NOT SET
10305 ;          5527      then
10306 ;          5528          begin
10307 ;          5529          CLR_MBUS;                      !CLEAR GO IF STILL SET
10308 ;          5530
10309 ;          5531          if .GO IS_SET then ERRDF (29, ASYNC, DUMPER) else ERRDF (30, SYNC, DUMPER);
10310 ;          5532
10311 ;          5533                                         !TST GO TO DETERMINE FAILING MOD
10312 ;          5534          PRINTB (FIV_FMT, WRD_1, PHR_2, WRD_11, FNC_5, WRD_19);
10313 ;          5535          end
10314 ;          5536      else                               !DRY NOT SET AND GO CLEARED
10315 ;          5537          begin                          !REPORT ERROR
10316 ;          5538          ERRDF (31, ASYNC, DUMPER);
10317 ;          5539          PRINTB (FIV_FMT, WRD_2, PHR_1, WRD_11, FNC_5, WRD_19);
10318 ;          5540          end;
10319 ;          5541
10320 ;          5542      DODU_FLG = ONE;                    !SET DODU_FLG
10321 ;          5543      end;
10322 ;          5544
10323 ;          5545  if .GO IS_SET                          !SEE IF GO CLEARED AFTER XFERR.
10324 ;          5546  then
10325 ;          5547      begin
10326 ;          5548      CLR_MBUS;                          !CLEAR GO IF STILL SET
10327 ;          5549
10328 ;          5550      if .GO IS_SET then ERRDF (32, ASYNC, DUMPER) else ERRDF (33, SYNC, DUMPER);
10329 ;          5551
10330 ;          5552                                         !TST GO TO DETERMINE FAILING MOD
10331 ;          5553      PRINTB (FIV_FMT, WRD_1, PHR_2, WRD_11, FNC_5, WRD_19);
10332 ;          5554      DODU_FLG = ONE;
10333 ;          5555      end;
10334 ;          5556
10335 ;          5557  ENDSUB;
10336 ;          5558
10337 ;          5559  if .TRE IS_SET                          !SEE IF XFERR CAUSED A TRANSFER ERROR
```

```
10339 :ML4                                                          14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
10340 :                                                             14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (64)
10341
10342 :          5560  then
10343 :          5561      begin                                    !REPORT ERROR IF SET AND CONTINUE TESTING
10344 :          5562      ERRDF (115, INTER, DUMPER);
10345 :          5563      PRINTB (SIX_FMT, WRD_61, WRD_20, PHR_5, WRD_12, FNC_5, WRD_19);
10346 :          5564      end;
10347 :          5565
10348 :          5566  if .DODU_FLG IS_SET                          !DROP THIS UNIT IF DODU_FLG SET
10349 :          5567  then
10350 :          5568      begin
10351 :          5569      DODU (.ML_LUN);
10352 :          5570      DOCLN;
10353 :          5571      end;
10354 :          5572
10355 :          5573  ENDTST;
10359
10360                                        .SBTTL  $T14
10364 035044  004167  146774          $T14: JSR     R1,$SAVE2                    ;                          5454
10365 035050  005746                        TST     -(SP)
10366 035052  104402                  1$:   TRAP    2                            ;                          5473
10367 035054  152777  000040  156132        BISB    #40,@ML.REG+40               ;                          5475
10368 035062  016701  156514              MOV     ML.DUT,R1
10369 035066  042701  177770              BIC     #177770,R1
10370 035072  142777  000007  156114        BICB    #7,@ML.REG+40
10371 035100  150177  156110              BISB    R1,@ML.REG+40
10372 035104  005002                       CLR     R2                           ; DODU.FLG                 5477
10373 035106  004767  160726              JSR     PC,FIRST.BLK.XFER            ;                          5478
10374 035112  012777  000061  156034        MOV     #61,@ML.REG                  ;                          5479
10375 035120  132777  000001  156026        BITB    #1,@ML.REG                   ;                          5481
10376 035126  001057                       BNE     2$
10377 035130  104455                       TRAP    55                           ;                          5484
10378 035132  000030                       .WORD   30
10379 035134  010464                       .WORD   ASYNC
10380 035136  024060                       .WORD   DUMPER
10381 035140  012746  006506              MOV     #WRD.19,-(SP)                ;                          5485
10382 035144  012746  007774              MOV     #FNC.5,-(SP)
10383 035150  012746  006426              MOV     #WRD.12,-(SP)
10384 035154  012746  007366              MOV     #PHR.1,-(SP)
10385 035160  012746  006276              MOV     #WRD.1,-(SP)
10386 035164  012746  006062              MOV     #FIV.FMT,-(SP)
10387 035170  012746  000006              MOV     #6,-(SP)
10388 035174  010600                       MOV     SP,R0                        ; SP,*
10389 035176  104414                       TRAP    14
10390 035200  132777  000200  156016        BITB    #200,@ML.REG+50              ;                          5487
10391 035206  001056                       BNE     3$
10392 035210  104455                       TRAP    55                           ;                          5490
10393 035212  000031                       .WORD   31
```

```
10395                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
10396                                   ;                                                       14-May-1981 13:05:08    PA:<
10397
10398 035214  010464                             .WORD   ASYNC
10399 035216  024060                             .WORD   DUMPER
10400 035220  012746  007504                     MOV     #PHR.6,-(SP)            ;                               5491
10401 035224  012746  006276                     MOV     #WRD.1,-(SP)
10402 035230  012746  006754                     MOV     #WRD.43,-(SP)
10403 035234  012746  007366                     MOV     #PHR.1,-(SP)
10404 035240  012746  006302                     MOV     #WRD.2,-(SP)
10405 035244  012746  006062                     MOV     #FIV.FMT,-(SP)
10406 035250  012746  000006                     MOV     #6,-(SP)
10407 035254  010600                            MOV     SP,R0                   ; SP,*
10408 035256  104414                            TRAP    14
10409 035260  062706  000016                     ADD     #16,SP                  ;                               5489
10410 035264  000427                            BR      3$                      ;                               5494
10411 035266  105777  155732          2$:       TSTB    @ML.REG+50              ;                               5498
10412 035272  100030                            BPL     4$
10413 035274  104455                            TRAP    55                      ;                               5501
10414 035276  000032                            .WORD   32
10415 035300  010464                            .WORD   ASYNC
10416 035302  024060                            .WORD   DUMPER
10417 035304  012746  007472                     MOV     #PHR.5,-(SP)            ;                               5502
10418 035310  012746  006276                     MOV     #WRD.1,-(SP)
10419 035314  012746  006754                     MOV     #WRD.43,-(SP)
10420 035320  012746  007404                     MOV     #PHR.2,-(SP)
10421 035324  012746  006302                     MOV     #WRD.2,-(SP)
10422 035330  012746  006062                     MOV     #FIV.FMT,-(SP)
10423 035334  012746  000006                     MOV     #6,-(SP)
10424 035340  010600                            MOV     SP,R0                   ; SP,*
10425 035342  104414                            TRAP    14
10426 035344  012702  000001          3$:       MOV     #1,R2                   ; *,DODU.FLG                    5503
10427 035350  062706  000016                     ADD     #16,SP                  ;                               5500
10428 035354  132777  000001  155652  4$:       BITB    #1,@ML.REG+60           ;                               5506
10429 035362  001426                            BEQ     5$
10430 035364  104455                            TRAP    55                      ;                               5509
10431 035366  000033                            .WORD   33
10432 035370  010464                            .WORD   ASYNC
10433 035372  024060                            .WORD   DUMPER
10434 035374  012746  006506                     MOV     #WRD.19,-(SP)           ;                               5510
10435 035400  012746  007774                     MOV     #FNC.5,-(SP)
10436 035404  012746  006426                     MOV     #WRD.12,-(SP)
10437 035410  012746  007472                     MOV     #PHR.5,-(SP)
10438 035414  012746  006314                     MOV     #WRD.3,-(SP)
10439 035420  012746  006062                     MOV     #FIV.FMT,-(SP)
10440 035424  012746  000006                     MOV     #6,-(SP)
10441 035430  010600                            MOV     SP,R0                   ; SP,*
10442 035432  104414                            TRAP    14
10443 035434  062706  000016                     ADD     #16,SP                  ;                               5508
10444 035440  032777  020000  155566  5$:       BIT     #20000,@ML.REG+60       ;                               5513
10445 035446  001426                            BEQ     6$
10446 035450  104455                            TRAP    55                      ;                               5516
10447 035452  000034                            .WORD   34
10448 035454  010464                            .WORD   ASYNC
10449 035456  024060                            .WORD   DUMPER
```

```
10451                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
10452                                   ;                                                       14-May-1981 13:05:08    PA:<
10453
10454 035460  012746  006506                    MOV     #WRD.19,-(SP)           ;                                       5517
10455 035464  012746  007774                    MOV     #FNC.5,-(SP)
10456 035470  012746  006426                    MOV     #WRD.12,-(SP)
10457 035474  012746  007472                    MOV     #PHR.5,-(SP)
10458 035500  012746  006322                    MOV     #WRD.4,-(SP)
10459 035504  012746  006062                    MOV     #FIV.FMT,-(SP)
10460 035510  012746  000006                    MOV     #6,-(SP)
10461 035514  010600                            MOV     SP,R0                   ; SP,*
10462 035516  104414                            TRAP    14
10463 035520  062706  000016                    ADD     #16,SP                  ;                                       5515
10464 035524  012700  000050            6$:     MOV     #50,R0                  ; *,$$TMP2                              5520
10465 035530  001410                    7$:     BEQ     10$
10466 035532  016701  144360                    MOV     LSDLY,R1                ; *,$$TMP1
10467 035536  001403                            BEQ     9$
10468 035540  005016                    8$:     CLR     (SP)                    ; $$TMP
10469 035542  005301                            DEC     R1                      ; $$TMP1
10470 035544  001375                            BNE     8$
10471 035546  005300                    9$:     DEC     R0                      ; $$TMP2
10472 035550  000767                            BR      7$
10473 035552  132777  000200  155444    10$:    BITB    #200,@ML.REG+50         ;                                       5522
10474 035560  001106                            BNE     15$
10475 035562  132777  000001  155364            BITB    #1,@ML.REG              ;                                       5526
10476 035570  001452                            BEQ     13$
10477 035572  152777  000040  155414            BISB    #40,@ML.REG+40          ;                                       5528
10478 035600  016701  155776                    MOV     ML.DUT,R1
10479 035604  042701  177770                    BIC     #177770,R1
10480 035610  142777  000007  155376            BICB    #7,@ML.REG+40
10481 035616  150177  155372                    BISB    R1,@ML.REG+40
10482 035622  132777  000001  155324            BITB    #1,@ML.REG              ;                                       5531
10483 035630  001405                            BEQ     11$
10484 035632  104455                            TRAP    55
10485 035634  000035                            .WORD   35
10486 035636  010464                            .WORD   ASYNC
10487 035640  024060                            .WORD   DUMPER
10488 035642  000404                            BR      12$
10489 035644  104455                    11$:    TRAP    55
10490 035646  000036                            .WORD   36
10491 035650  010526                            .WORD   SYNC
10492 035652  024060                            .WORD   DUMPER
10493 035654  012746  006506            12$:    MOV     #WRD.19,-(SP)           ;                                       5534
10494 035660  012746  007774                    MOV     #FNC.5,-(SP)
10495 035664  012746  006416                    MOV     #WRD.11,-(SP)
10496 035670  012746  007404                    MOV     #PHR.2,-(SP)
10497 035674  012746  006276                    MOV     #WRD.1,-(SP)
10498 035700  012746  006062                    MOV     #FIV.FMT,-(SP)
10499 035704  012746  000006                    MOV     #6,-(SP)
10500 035710  010600                            MOV     SP,R0                   ; SP,*
10501 035712  104414                            TRAP    14
10502 035714  000424                            BR      14$                     ;                                       5526
10503 035716  104455                    13$:    TRAP    55                      ;                                       5538
10504 035720  000037                            .WORD   37
10505 035722  010464                            .WORD   ASYNC
```

```
10507                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
10508                                   ;                                                       14-May-1981 13:05:08    PA:<
10509
10510 035724  024060                            .WORD   DUMPER
10511 035726  012746  006506                    MOV     #WRD.19,-(SP)           ;                               5539
10512 035732  012746  007774                    MOV     #FNC.5,-(SP)
10513 035736  012746  006416                    MOV     #WRD.11,-(SP)
10514 035742  012746  007366                    MOV     #PHR.1,-(SP)
10515 035746  012746  006302                    MOV     #WRD.2,-(SP)
10516 035752  012746  006062                    MOV     #FIV.FMT,-(SP)
10517 035756  012746  000006                    MOV     #6,-(SP)
10518 035762  010600                            MOV     SP,R0                   ; SP,*
10519 035764  104414                            TRAP    14
10520 035766  012702  000001            14$:    MOV     #1,R2                   ; *,DODU.FLG                    5542
10521 035772  062706  000016                    ADD     #16,SP                  ;                               5524
10522 035776  132777  000001  155150    15$:    BITB    #1,@ML.REG              ;                               5545
10523 036004  001455                            BEQ     18$
10524 036006  152777  000040  155200            BISB    #40,@ML.REG+40          ;                               5547
10525 036014  016701  155562                    MOV     ML.DUT,R1
10526 036020  042701  177770                    BIC     #177770,R1
10527 036024  142777  000007  155162            BICB    #7,@ML.REG+40
10528 036032  150177  155156                    BISB    R1,@ML.REG+40
10529 036036  132777  000001  155110            BITB    #1,@ML.REG              ;                               5550
10530 036044  001405                            BEQ     16$
10531 036046  104455                            TRAP    55
10532 036050  000040                            .WORD   40
10533 036052  010464                            .WORD   ASYNC
10534 036054  024060                            .WORD   DUMPER
10535 036056  000404                            BR      17$
10536 036060  104455            16$:    TRAP    55
10537 036062  000041                            .WORD   41
10538 036064  010526                            .WORD   SYNC
10539 036066  024060                            .WORD   DUMPER
10540 036070  012746  006506            17$:    MOV     #WRD.19,-(SP)           ;                               5553
10541 036074  012746  007774                    MOV     #FNC.5,-(SP)
10542 036100  012746  006416                    MOV     #WRD.11,-(SP)
10543 036104  012746  007404                    MOV     #PHR.2,-(SP)
10544 036110  012746  006276                    MOV     #WRD.1,-(SP)
10545 036114  012746  006062                    MOV     #FIV.FMT,-(SP)
10546 036120  012746  000006                    MOV     #6,-(SP)
10547 036124  010600                            MOV     SP,R0                   ; SP,*
10548 036126  104414                            TRAP    14
10549 036130  012702  000001                    MOV     #1,R2                   ; *,DODU.FLG                    5554
10550 036134  062706  000016                    ADD     #16,SP                  ;                               5547
10551 036140  104467            18$:    TRAP    67                      ;                               5555
10552 036142  006000                            ROR     R0
10553 036144  103002                            BHIS    19$
10554 036146  000167  176700                    JMP     1$
10555 036152  032777  040000  154774    19$:    BIT     #40000,@ML.REG          ;                               5559
10556 036160  001430                            BEQ     20$
10557 036162  104455                            TRAP    55                      ;                               5562
10558 036164  000163                            .WORD   163
10559 036166  010672                            .WORD   INTER
10560 036170  024060                            .WORD   DUMPER
10561 036172  012746  006506                    MOV     #WRD.19,-(SP)           ;                               5563
```

```
10563                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
10564                                   ;                                                       14-May-1981 13:05:08    PA:<
10565
10566 036176  012746  007774                     MOV     #FNC.5,-(SP)
10567 036202  012746  006426                     MOV     #WRD.12,-(SP)
10568 036206  012746  007472                     MOV     #PHR.5,-(SP)
10569 036212  012746  006514                     MOV     #WRD.20,-(SP)
10570 036216  012746  007174                     MOV     #WRD.61,-(SP)
10571 036222  012746  006100                     MOV     #SIX.FMT,-(SP)
10572 036226  012746  000007                     MOV     #7,-(SP)
10573 036232  010600                            MOV     SP,R0                           ; SP,*
10574 036234  104414                            TRAP    14
10575 036236  062706  000020                     ADD     #20,SP                          ;                                       5561
10576 036242  005302                  20$:       DEC     R2                              ; DODU.FLG                              5566
10577 036244  001004                            BNE     21$
10578 036246  016700  155326                     MOV     ML.LUN,R0                       ;                                       5569
10579 036252  104451                            TRAP    51
10580 036254  104444                            TRAP    44
10581 036256  005726                  21$:       TST     (SP)+                           ;                                       5454
10582 036260  000207                            RTS     PC
10583
10584                                   ; Routine Size:  327 words
10585                                   ; Maximum stack depth per invocation:  18 words
10590
10591
10595
10596                                           .SBTTL  T14
10600 036262                          T14::
10601 036262  004767  176556          1$:       JSR     PC,$T14                         ;                                       5571
10602 036266  104466                            TRAP    66
10603 036270  006000                            ROR     R0
10604 036272  103773                            BLO     1$
10605 036274  000207                            RTS     PC
10606
10607                                   ; Routine Size:  6 words
10608                                   ; Maximum stack depth per invocation:  0 words
10613
10614
10615 ;           5574  !<BLF/PAGE>
```

```
10617 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
10618 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (65)
10619
10620 ;          5575  !
10621 ;          5576  !
10622 ;          5577  BGNTST;
10623 ;          5578
10624 ;          5579  !++
10625 ;          5580  ! TEST NUMBER:  TST 15
10626 ;          5581  !
10627 ;          5582  ! TEST NAME:    READ FUNCTION TEST
10628 ;          5583  !
10629 ;          5584  ! TEST DESCRIPTION:
10630 ;          5585  !
10631 ;          5586  !       TEST IF THE DRIVE CAN PERFORM
10632 ;          5587  !       A READ FUNCTION WITHOUT
10633 ;          5588  !       HANGING THE DRIVE.
10634 ;          5589  !
10635 ;          5590  !       A READ FUNCTION IS WRITTEN
10636 ;          5591  !       INTO MLCS1
10637 ;          5592  !
10638 ;          5593  !       THEN GO AND ERROR BITS ARE
10639 ;          5594  !       CHECKED FOR CORRECT STATES.
10640 ;          5595  !
10641 ;          5596  !--
10642 ;          5597
10643 ;          5598  local
10644 ;          5599      DODU_FLG;                                !DROP UNIT FLAG
10645 ;          5600
10646 ;          5601  BGNSUB;
10647 ;          5602  CLR_MBUS;
10648 ;          5603  DODU_FLG = ZERO;
10649 ;          5604  FIRST_BLK_XFER ();                           !SET UP A FIRST BLK XFERR
10650 ;          5605  MLCS1 = read;                                !DO A READ FUNCTION
10651 ;          5606
10652 ;          5607  if .GO IS_NOT_SET                            !SEE IF GO GOT SET
10653 ;          5608  then
10654 ;          5609      begin                                    !ERROR IF CLEAR
10655 ;          5610      ERRDF (34, ASYNC, DUMPER);
10656 ;          5611      PRINTB (FIV_FMT, WRD_1, PHR_1, WRD_12, FNC_6, WRD_19);
10657 ;          5612
10658 ;          5613      if .DRY IS_NOT_SET                       !TST DRY SET WITH GO CLEAR
10659 ;          5614      then
10660 ;          5615          begin                                !ERROR IF NOT SET
10661 ;          5616          ERRDF (35, ASYNC, DUMPER);
10662 ;          5617          PRINTB (FIV_FMT, WRD_2, PHR_1, WRD_43, WRD_1, PHR_6);
10663 ;          5618          end;
10664 ;          5619
10665 ;          5620      DODU_FLG = ONE;                          !SET DODU_FLG
10666 ;          5621      end
10667 ;          5622  else                                         !GO BIT GOT SET
10668 ;          5623
10669 ;          5624      if .DRY IS_SET                           !SEE IF DRY IS CLEAR
10670 ;          5625      then
10671 ;          5626          begin                                !ERROR IF SET
10672 ;ML4                                                        14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
10673 ;                                                           14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (65)
```

```
10674
10675 ;          5627          ERRDF (36, ASYNC, DUMPER);
10676 ;          5628          PRINTB (FIV_FMT, WRD_2, PHR_2, WRD_43, WRD_1, PHR_5);
10677 ;          5629          DODU_FLG = ONE;
10678 ;          5630          end;
10679 ;          5631
10680 ;          5632  if .ILF IS_SET                                  !DID FUNCTION CAUSE ILF
10681 ;          5633  then
10682 ;          5634      begin                                       !ERROR IF YES
10683 ;          5635      ERRDF (37, ASYNC, DUMPER);
10684 ;          5636      PRINTB (FIV_FMT, WRD_3, PHR_5, WRD_12, FNC_6, WRD_19);
10685 ;          5637      end;
10686 ;          5638
10687 ;          5639  if .OPI IS_SET                                  !DID FUNCTION CAUSE OPI
10688 ;          5640  then
10689 ;          5641      begin                                       !ERROR IF YES
10690 ;          5642      ERRDF (38, ASYNC, DUMPER);
10691 ;          5643      PRINTB (FIV_FMT, WRD_4, PHR_5, WRD_12, FNC_6, WRD_19);
10692 ;          5644      end;
10693 ;          5645
10694 ;          5646  DELAY (FRTY_US);                                !WAIT FOR XFERR TO COMPLETE
10695 ;          5647
10696 ;          5648  if .DRY IS_NOT_SET                              !IS DRY SET AFTER XFERR
10697 ;          5649  then
10698 ;          5650      begin
10699 ;          5651
10700 ;          5652      if .GO IS_SET                               !TEST GO CLEAR WITH DRY NOT SET
10701 ;          5653      then
10702 ;          5654          begin
10703 ;          5655          CLR_MBUS;                               !CLEAR GO
10704 ;          5656
10705 ;          5657          if .GO IS_SET then ERRDF (39, ASYNC, DUMPER) else ERRDF (40, SYNC, DUMPER);
10706 ;          5658
10707 ;          5659                                                  !TST GO TO DETERMINE FAILING MOD
10708 ;          5660          PRINTB (FIV_FMT, WRD_1, PHR_2, WRD_11, FNC_6, WRD_19);
10709 ;          5661          end
10710 ;          5662      else                                        !GO BIT CLEAR WITH DRY NOT SET
10711 ;          5663          begin                                   !REPORT ERROR
10712 ;          5664          ERRDF (41, ASYNC, DUMPER);
10713 ;          5665          PRINTB (FIV_FMT, WRD_2, PHR_1, WRD_11, FNC_6, WRD_19);
10714 ;          5666          end;
10715 ;          5667
10716 ;          5668      DODU_FLG = ONE;                             !SET DODU_FLG
10717 ;          5669      end;
10718 ;          5670
10719 ;          5671  if .GO IS_SET                                   !SEE IF GO CLEAR AFTER XFERR
10720 ;          5672  then
10721 ;          5673      begin
10722 ;          5674      CLR_MBUS;                                   !CLEAR GO IF STILL SET
10723 ;          5675
10724 ;          5676      if .GO IS_SET then ERRDF (42, ASYNC, DUMPER) else ERRDF (43, SYNC, DUMPER);
10725 ;          5677
10726 ;          5678                                                  !TST GO TO DETERMINE FAILING MOD
10727 ;ML4                                                            14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
10728 ;                                                              14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (65)
10729
10730 ;          5679      PRINTB (FIV_FMT, WRD_1, PHR_2, WRD_11, FNC_6, WRD_19);
```

```
 10731 ;          5680      DODU_FLG = ONE;
 10732 ;          5681      end;
 10733 ;          5682
 10734 ;          5683  ENDSUB;
 10735 ;          5684
 10736 ;          5685  if .TRE IS_SET                              !SEE IF XFERR CAUSED A TRANSFER ERROR
 10737 ;          5686  then
 10738 ;          5687      begin                                   !REPORT ERROR IF SET AND CONTINUE TESTING
 10739 ;          5688      ERRDF (116, INTER, DUMPER);
 10740 ;          5689      PRINTB (SIX_FMT, WRD_61, WRD_20, PHR_5, WRD_12, FNC_6, WRD_19);
 10741 ;          5690      DODU_FLG = ONE;
 10742 ;          5691      end;
 10743 ;          5692
 10744 ;          5693  if .DODU_FLG IS_SET                         !DROP THIS UNIT IF DODU_FLG SET
 10745 ;          5694  then
 10746 ;          5695      begin
 10747 ;          5696      DODU (.ML_LUN);
 10748 ;          5697      DOCLN;
 10749 ;          5698      end;
 10750 ;          5699
 10751 ;          5700  ENDTST;
 10755
 10756                                        .SBTTL  $T15
 10760 036276  004167  145542          $T15: JSR     R1,$SAVE2                 ;                    5573
 10761 036302  005746                        TST     -(SP)
 10762 036304  104402                  1$:   TRAP    2                         ;                    5599
 10763 036306  152777  000040  154700        BISB    #40,@ML.REG+40            ;                    5601
 10764 036314  016701  155262                MOV     ML.DUT,R1
 10765 036320  042701  177770                BIC     #177770,R1
 10766 036324  142777  000007  154662        BICB    #7,@ML.REG+40
 10767 036332  150177  154656                BISB    R1,@ML.REG+40
 10768 036336  005002                        CLR     R2                        ; DODU.FLG           5603
 10769 036340  004767  157474                JSR     PC,FIRST.BLK.XFER         ;                    5604
 10770 036344  012777  000071  154602        MOV     #71,@ML.REG               ;                    5605
 10771 036352  132777  000001  154574        BITB    #1,@ML.REG                ;                    5607
 10772 036360  001057                        BNE     2$
 10773 036362  104455                        TRAP    55                        ;                    5610
 10774 036364  000042                        .WORD   42
 10775 036366  010464                        .WORD   ASYNC
 10776 036370  024060                        .WORD   DUMPER
 10777 036372  012746  006506                MOV     #WRD.19,-(SP)             ;                    5611
 10778 036376  012746  010004                MOV     #FNC.6,-(SP)
 10779 036402  012746  006426                MOV     #WRD.12,-(SP)
 10780 036406  012746  007366                MOV     #PHR.1,-(SP)
 10781 036412  012746  006276                MOV     #WRD.1,-(SP)
 10782                               ;ML4                                14-May-1981 13:48:54  TOPS
 10783                               ;                                   14-May-1981 13:05:08  PA:<
 10784
 10785 036416  012746  006062                MOV     #FIV.FMT,-(SP)
 10786 036422  012746  000006                MOV     #6,-(SP)
 10787 036426  010600                        MOV     SP,R0                     ; SP,*
 10788 036430  104414                        TRAP    14
 10789 036432  132777  000200  154564        BITB    #200,@ML.REG+50           ;                    5613
 10790 036440  001056                        BNE     3$
 10791 036442  104455                        TRAP    55                        ;                    5616
 10792 036444  000043                        .WORD   43
 10793 036446  010464                        .WORD   ASYNC
```

```
10794 036450  024060                        .WORD   DUMPER
10795 036452  012746  007504                MOV     #PHR.6,-(SP)        ;                                   5617
10796 036456  012746  006276                MOV     #WRD.1,-(SP)
10797 036462  012746  006754                MOV     #WRD.43,-(SP)
10798 036466  012746  007366                MOV     #PHR.1,-(SP)
10799 036472  012746  006302                MOV     #WRD.2,-(SP)
10800 036476  012746  006062                MOV     #FIV.FMT,-(SP)
10801 036502  012746  000006                MOV     #6,-(SP)
10802 036506  010600                        MOV     SP,R0               ; SP,*
10803 036510  104414                        TRAP    14
10804 036512  062706  000016                ADD     #16,SP              ;                                   5615
10805 036516  000427                        BR      3$                  ;                                   5620
10806 036520  105777  154500          2$:   TSTB    @ML.REG+50          ;                                   5624
10807 036524  100030                        BPL     4$
10808 036526  104455                        TRAP    55                  ;                                   5627
10809 036530  000044                        .WORD   44
10810 036532  010464                        .WORD   ASYNC
10811 036534  024060                        .WORD   DUMPER
10812 036536  012746  007472                MOV     #PHR.5,-(SP)        ;                                   5628
10813 036542  012746  006276                MOV     #WRD.1,-(SP)
10814 036546  012746  006754                MOV     #WRD.43,-(SP)
10815 036552  012746  007404                MOV     #PHR.2,-(SP)
10816 036556  012746  006302                MOV     #WRD.2,-(SP)
10817 036562  012746  006062                MOV     #FIV.FMT,-(SP)
10818 036566  012746  000006                MOV     #6,-(SP)
10819 036572  010600                        MOV     SP,R0               ; SP,*
10820 036574  104414                        TRAP    14
10821 036576  012702  000001          3$:   MOV     #1,R2               ; *,DODU.FLG                        5629
10822 036602  062706  000016                ADD     #16,SP              ;                                   5626
10823 036606  132777  000001  154420  4$:   BITB    #1,@ML.REG+60       ;                                   5632
10824 036614  001426                        BEQ     5$
10825 036616  104455                        TRAP    55                  ;                                   5635
10826 036620  000045                        .WORD   45
10827 036622  010464                        .WORD   ASYNC
10828 036624  024060                        .WORD   DUMPER
10829 036626  012746  006506                MOV     #WRD.19,-(SP)       ;                                   5636
10830 036632  012746  010004                MOV     #FNC.6,-(SP)
10831 036636  012746  006426                MOV     #WRD.12,-(SP)
10832 036642  012746  007472                MOV     #PHR.5,-(SP)
10833 036646  012746  006314                MOV     #WRD.3,-(SP)
10834 036652  012746  006062                MOV     #FIV.FMT,-(SP)
10835 036656  012746  000006                MOV     #6,-(SP)
10836 036662  010600                        MOV     SP,R0               ; SP,*
10837                             ;ML4                                  14-May-1981 13:48:54    TOPS
10838                             ;                                     14-May-1981 13:05:08    PA:<
10839
10840 036664  104414                        TRAP    14
10841 036666  062706  000016                ADD     #16,SP              ;                                   5634
10842 036672  032777  020000  154334  5$:   BIT     #20000,@ML.REG+60   ;                                   5639
10843 036700  001426                        BEQ     6$
10844 036702  104455                        TRAP    55                  ;                                   5642
10845 036704  000046                        .WORD   46
10846 036706  010464                        .WORD   ASYNC
10847 036710  024060                        .WORD   DUMPER
10848 036712  012746  006506                MOV     #WRD.19,-(SP)       ;                                   5643
10849 036716  012746  010004                MOV     #FNC.6,-(SP)
10850 036722  012746  006426                MOV     #WRD.12,-(SP)
```

```
10851 036726  012746  007472                    MOV     #PHR.5,-(SP)
10852 036732  012746  006322                    MOV     #WRD.4,-(SP)
10853 036736  012746  006062                    MOV     #FIV.FMT,-(SP)
10854 036742  012746  000006                    MOV     #6,-(SP)
10855 036746  010600                            MOV     SP,R0                   ; SP,*
10856 036750  104414                            TRAP    14
10857 036752  062706  000016                    ADD     #16,SP                  ;                                       5641
10858 036756  012700  000050            6$:     MOV     #50,R0                  ; *,$$TMP2                              5646
10859 036762  001410            7$:     BEQ     10$
10860 036764  016701  143126                    MOV     L$DLY,R1                ; *,$$TMP1
10861 036770  001403                            BEQ     9$
10862 036772  005016            8$:     CLR     (SP)                    ; $$TMP
10863 036774  005301                            DEC     R1                      ; $$TMP1
10864 036776  001375                            BNE     8$
10865 037000  005300            9$:     DEC     R0                      ; $$TMP2
10866 037002  000767                            BR      7$
10867 037004  132777  000200  154212    10$:    BITB    #200,@ML.REG+50         ;                                       5648
10868 037012  001106                            BNE     15$
10869 037014  132777  000001  154132            BITB    #1,@ML.REG              ;                                       5652
10870 037022  001452                            BEQ     13$
10871 037024  152777  000040  154162            BISB    #40,@ML.REG+40          ;                                       5654
10872 037032  016701  154544                    MOV     ML.DUT,R1
10873 037036  042701  177770                    BIC     #177770,R1
10874 037042  142777  000007  154144            BICB    #7,@ML.REG+40
10875 037050  150177  154140                    BISB    R1,@ML.REG+40
10876 037054  132777  000001  154072            BITB    #1,@ML.REG              ;                                       5657
10877 037062  001405                            BEQ     11$
10878 037064  104455                            TRAP    55
10879 037066  000047                            .WORD   47
10880 037070  010464                            .WORD   ASYNC
10881 037072  024060                            .WORD   DUMPER
10882 037074  000404                            BR      12$
10883 037076  104455            11$:    TRAP    55
10884 037100  000050                            .WORD   50
10885 037102  010526                            .WORD   SYNC
10886 037104  024060                            .WORD   DUMPER
10887 037106  012746  006506    12$:    MOV     #WRD.19,-(SP)           ;                                       5660
10888 037112  012746  010004                    MOV     #FNC.6,-(SP)
10889 037116  012746  006416                    MOV     #WRD.11,-(SP)
10890 037122  012746  007404                    MOV     #PHR.2,-(SP)
10891 037126  012746  006276                    MOV     #WRD.1,-(SP)
10892                         ;ML4                                                    14-May-1981 13:48:54    TOPS
10893                         ;                                                       14-May-1981 13:05:08    PA:<
10894
10895 037132  012746  006062                    MOV     #FIV.FMT,-(SP)
10896 037136  012746  000006                    MOV     #6,-(SP)
10897 037142  010600                            MOV     SP,R0                   ; SP,*
10898 037144  104414                            TRAP    14
10899 037146  000424                            BR      14$                     ;                                       5652
10900 037150  104455            13$:    TRAP    55                      ;                                       5664
10901 037152  000051                            .WORD   51
10902 037154  010464                            .WORD   ASYNC
10903 037156  024060                            .WORD   DUMPER
10904 037160  012746  006506                    MOV     #WRD.19,-(SP)           ;                                       5665
10905 037164  012746  010004                    MOV     #FNC.6,-(SP)
10906 037170  012746  006416                    MOV     #WRD.11,-(SP)
10907 037174  012746  007366                    MOV     #PHR.1,-(SP)
```

```
10908 037200  012746  006302                MOV     #WRD.2,-(SP)
10909 037204  012746  006062                MOV     #FIV.FMT,-(SP)
10910 037210  012746  000006                MOV     #6,-(SP)
10911 037214  010600                        MOV     SP,R0                   ; SP,*
10912 037216  104414                        TRAP    14
10913 037220  012702  000001          14$:  MOV     #1,R2                   ; *,DODU.FLG                    5668
10914 037224  062706  000016                ADD     #16,SP                  ;                               5650
10915 037230  132777  000001  153716  15$:  BITB    #1,@ML.REG              ;                               5671
10916 037236  001455                        BEQ     18$
10917 037240  152777  000040  153746        BISB    #40,@ML.REG+40          ;                               5673
10918 037246  016701  154330                MOV     ML.DUT,R1
10919 037252  042701  177770                BIC     #177770,R1
10920 037256  142777  000007  153730        BICB    #7,@ML.REG+40
10921 037264  150177  153724                BISB    R1,@ML.REG+40
10922 037270  132777  000001  153656        BITB    #1,@ML.REG              ;                               5676
10923 037276  001405                        BEQ     16$
10924 037300  104455                        TRAP    55
10925 037302  000052                        .WORD   52
10926 037304  010464                        .WORD   ASYNC
10927 037306  024060                        .WORD   DUMPER
10928 037310  000404                        BR      17$
10929 037312  104455          16$:  TRAP    55
10930 037314  000053                        .WORD   53
10931 037316  010526                        .WORD   SYNC
10932 037320  024060                        .WORD   DUMPER
10933 037322  012746  006506          17$:  MOV     #WRD.19,-(SP)           ;                               5679
10934 037326  012746  010004                MOV     #FNC.6,-(SP)
10935 037332  012746  006416                MOV     #WRD.11,-(SP)
10936 037336  012746  007404                MOV     #PHR.2,-(SP)
10937 037342  012746  006276                MOV     #WRD.1,-(SP)
10938 037346  012746  006062                MOV     #FIV.FMT,-(SP)
10939 037352  012746  000006                MOV     #6,-(SP)
10940 037356  010600                        MOV     SP,R0                   ; SP,*
10941 037360  104414                        TRAP    14
10942 037362  012702  000001                MOV     #1,R2                   ; *,DODU.FLG                    5680
10943 037366  062706  000016                ADD     #16,SP                  ;                               5673
10944 037372  104467          18$:  TRAP    67                              ;                               5681
10945 037374  006000                        ROR     R0
10946 037376  103002                        BHIS    19$
10947                       ;ML4                                                    14-May-1981 13:48:54    TOPS
10948                       ;                                                       14-May-1981 13:05:08    PA:<
10949
10950 037400  000167  176700                JMP     1$
10951 037404  032777  040000  153542  19$:  BIT     #40000,@ML.REG          ;                               5685
10952 037412  001432                        BEQ     20$
10953 037414  104455                        TRAP    55                      ;                               5688
10954 037416  000164                        .WORD   164
10955 037420  010672                        .WORD   INTER
10956 037422  024060                        .WORD   DUMPER
10957 037424  012746  006506                MOV     #WRD.19,-(SP)           ;                               5689
10958 037430  012746  010004                MOV     #FNC.6,-(SP)
10959 037434  012746  006426                MOV     #WRD.12,-(SP)
10960 037440  012746  007472                MOV     #PHR.5,-(SP)
10961 037444  012746  006514                MOV     #WRD.20,-(SP)
10962 037450  012746  007174                MOV     #WRD.61,-(SP)
10963 037454  012746  006100                MOV     #SIX.FMT,-(SP)
10964 037460  012746  000007                MOV     #7,-(SP)
```

```
10965 037464  010600                    MOV     SP,R0                           ; SP,*
10966 037466  104414                    TRAP    14
10967 037470  012702  000001            MOV     #1,R2                           ; *,DODU.FLG                    5690
10968 037474  062706  000020            ADD     #20,SP                          ;                               5687
10969 037500  005302              20$:  DEC     R2                              ; DODU.FLG                      5693
10970 037502  001004                    BNE     21$
10971 037504  016700  154070            MOV     ML.LUN,R0                       ;                               5696
10972 037510  104451                    TRAP    51
10973 037512  104444                    TRAP    44
10974 037514  005726              21$:  TST     (SP)+                           ;                               5573
10975 037516  000207                    RTS     PC
10976
10977                             ; Routine Size:  329 words
10978                             ; Maximum stack depth per invocation:  18 words
10983
10984
10988
10989                                     .SBTTL  T15
10993 037520                      T15::
10994 037520  004767  176552      1$:   JSR     PC,$T15                         ;                               5698
10995 037524  104466                    TRAP    66
10996 037526  006000                    ROR     R0
10997 037530  103773                    BLO     1$
10998 037532  000207                    RTS     PC
10999
11000                             ; Routine Size:  6 words
11001                             ; Maximum stack depth per invocation:  0 words
11002                             ;ML4                                                    14-May-1981 13:48:54    TOPS
11003                             ;                                                       14-May-1981 13:05:08    PA:<
11004
11009
11010
11011 ;           5701   !<BLF/PAGE>
```

```
11013 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
11014 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (66)
11015
11016 ;          5702  !
11017 ;          5703  BGNTST;
11018 ;          5704
11019 ;          5705  !++
11020 ;          5706  ! TEST NUMBER:  TST 16
11021 ;          5707  !
11022 ;          5708  ! TEST NAME:    CLEAR FUNCTION TEST
11023 ;          5709  !
11024 ;          5710  ! TEST DESCRIPTION:
11025 ;          5711  !       TEST IF THE DRIVE CAN PERFORM A CLEAR FUNCTION WITHOUT HANGING THE DRIVE.
11026 ;          5712  !
11027 ;          5713  !       A CLEAR FUNCTION IS WRITTEN INTO MLCS1.
11028 ;          5714  !
11029 ;          5715  !       THEN GO AND ERROR BITS ARE CHECKED FOR CORRECT STATUS.
11030 ;          5716  !       THIS DRIVE IS DROPPED ON DETECTED ERRORS.
11031 ;          5717  !
11032 ;          5718  !--
11033 ;          5719
11034 ;          5720  local
11035 ;          5721      DODU_FLG;                                !DROP UNIT FLAG
11036 ;          5722
11037 ;          5723  BGNSUB;
11038 ;          5724  CLR_MBUS;
11039 ;          5725  DODU_FLG = ZERO;
11040 ;          5726  MLER = ONES;                                 !SET BITS IN ERROR REGISTER
11041 ;          5727  MLCS1 = DRV_CLR;                             !DO A CLEAR FUNCTION
11042 ;          5728  DELAY (ONE_DS);                              !DELAY
11043 ;          5729
11044 ;          5730  if .GO IS_SET                                !SEE IF GO CLEARED AFTER FUNCTION
11045 ;          5731  then
11046 ;          5732      begin                                    !ERROR IF SET
11047 ;          5733      ERRDF (44, ASYNC, DUMPER);
11048 ;          5734      PRINTB (FIV_FMT, WRD_1, PHR_5, WRD_11, FNC_3, FNC_7, WRD_19);
11049 ;          5735
11050 ;          5736      if .DRY IS_SET                           !TST DRY CLEAR WITH GO SET
11051 ;          5737      then
11052 ;          5738          begin                                !ERROR IF SET
11053 ;          5739          ERRDF (45, ASYNC, DUMPER);
11054 ;          5740          PRINTB (FIV_FMT, WRD_2, PHR_5, WRD_43, WRD_1, PHR_5);
11055 ;          5741          end;
11056 ;          5742
11057 ;          5743      DODU_FLG = ONE;                          !SET DODU_FLG
11058 ;          5744      end
11059 ;          5745  else                                         !GO CLEARED AFTER FUNCTION
11060 ;          5746
11061 ;          5747      if .DRY IS_NOT_SET                       !TST DRY SET WITH GO CLEAR
11062 ;          5748      then
11063 ;          5749          begin                                !ERROR IF NOT SET
11064 ;          5750          ERRDF (46, ASYNC, DUMPER);
11065 ;          5751          PRINTB (FIV_FMT, WRD_2, PHR_1, WRD_43, WRD_1, PHR_6);
11066 ;          5752          DODU_FLG = ONE;
11067 ;          5753          end;
```

```
11069 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
11070 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (66)
11071
11072 ;          5754
11073 ;          5755  if .ILF IS_SET                                !DID FUNCTION CAUSE ILF
11074 ;          5756  then
11075 ;          5757      begin                                     !ERROR IF YES
11076 ;          5758      ERRDF (47, ASYNC, DUMPER);
11077 ;          5759      PRINTB (FIV_FMT, WRD_3, PHR_5, WRD_12, FNC_3, FNC_7, WRD_19);
11078 ;          5760      end;
11079 ;          5761
11080 ;          5762  if .OPI IS_SET                                !DID FUNCTION CAUSE OPI
11081 ;          5763  then
11082 ;          5764      begin                                     !ERROR IF YES
11083 ;          5765      ERRDF (48, ASYNC, DUMPER);
11084 ;          5766      PRINTB (FIV_FMT, WRD_4, PHR_5, WRD_12, FNC_3, FNC_7, WRD_19);
11085 ;          5767      end;
11086 ;          5768
11087 ;          5769  if .MLER neq ZERO                             !TEST ERROR REGISTER FOR CLEAR
11088 ;          5770  then
11089 ;          5771      begin                                     !ERROR IF NOT CLEAR
11090 ;          5772      ERRDF (49, ASYNC, DUMPER);
11091 ;          5773      PRINTB (SIX_FMT, FNC_3, FNC_7, WRD_19, WRD_14, WRD_13, REG_3);
11092 ;          5774      end;
11093 ;          5775
11094 ;          5776  ENDSUB;
11095 ;          5777
11096 ;          5778  if .DODU_FLG IS_SET                           !DROP THIS UNIT IF DODU_FLG SET
11097 ;          5779  then
11098 ;          5780      begin
11099 ;          5781      DODU (.ML_LUN);
11100 ;          5782      DOCLN;
11101 ;          5783      end;
11102 ;          5784
11103 ;          5785  ENDTST;
11107
11108                                        .SBTTL  $T16
11112 037534  004167  144304          $T16: JSR     R1,$SAVE2                 ;                          5700
11113 037540  005746                        TST     -(SP)
11114 037542  104402                  1$:   TRAP    2                         ;                          5721
11115 037544  152777  000040  153442        BISB    #40,@ML.REG+40            ;                          5723
11116 037552  016701  154024              MOV     ML.DUT,R1
11117 037556  042701  177770              BIC     #177770,R1
11118 037562  142777  000007  153424        BICB    #7,@ML.REG+40
11119 037570  150177  153420              BISB    R1,@ML.REG+40
11120 037574  005002                      CLR     R2                        ; DODU.FLG                 5725
11121 037576  012777  177777  153430        MOV     #-1,@ML.REG+60            ;                          5726
11122 037604  012777  000011  153342        MOV     #11,@ML.REG               ;                          5727
11123 037612  012700  000001              MOV     #1,R0                     ; *,$$TMP2                 5728
```

```
11125                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
11126                                   ;                                                       14-May-1981 13:05:08    PA:<
11127
11128 037616  001410                    2$:     BEQ     5$
11129 037620  016701  142272                    MOV     L$DLY,R1        ; *,$$TMP1
11130 037624  001403                            BEQ     4$
11131 037626  005016                    3$:     CLR     (SP)            ; $$TMP
11132 037630  005301                            DEC     R1              ; $$TMP1
11133 037632  001375                            BNE     3$
11134 037634  005300                    4$:     DEC     R0              ; $$TMP2
11135 037636  000767                            BR      2$
11136 037640  132777  000001  153306    5$:     BITB    #1,@ML.REG      ;                       5730
11137 037646  001464                            BEQ     7$
11138 037650  104455                            TRAP    55              ;                       5733
11139 037652  000054                            .WORD   54
11140 037654  010464                            .WORD   ASYNC
11141 037656  024060                            .WORD   DUMPER
11142 037660  012746  006506                    MOV     #WRD.19,-(SP)   ;                       5734
11143 037664  ^12746  010012                    MOV     #FNC.7,-(SP)
11144 037670  012746  007750                    MOV     #FNC.3,-(SP)
11145 037674  012746  006416                    MOV     #WRD.11,-(SP)
11146 037700  012746  007472                    MOV     #PHR.5,-(SP)
11147 037704  012746  006276                    MOV     #WRD.1,-(SP)
11148 037710  012746  006062                    MOV     #FIV.FMT,-(SP)
11149 037714  012746  000007                    MOV     #7,-(SP)
11150 037720  010600                            MOV     SP,R0           ; SP,*
11151 037722  104414                            TRAP    14
11152 037724  105777  153274                    TSTB    @ML.REG+50      ;                       5736
11153 037730  100026                            BPL     6$
11154 037732  104455                            TRAP    55              ;                       5739
11155 037734  000055                            .WORD   55
11156 037736  010464                            .WORD   ASYNC
11157 037740  024060                            .WORD   DUMPER
11158 037742  012746  007472                    MOV     #PHR.5,-(SP)    ;                       5740
11159 037746  012746  006276                    MOV     #WRD.1,-(SP)
11160 037752  012746  006754                    MOV     #WRD.43,-(SP)
11161 037756  012746  007472                    MOV     #PHR.5,-(SP)
11162 037762  012746  006302                    MOV     #WRD.2,-(SP)
11163 037766  012746  006062                    MOV     #FIV.FMT,-(SP)
11164 037772  012746  000006                    MOV     #6,-(SP)
11165 037776  010600                            MOV     SP,R0           ; SP,*
11166 040000  104414                            TRAP    14
11167 040002  062706  000016                    ADD     #16,SP          ;                       5738
11168 040006  012702  000001            6$:     MOV     #1,R2           ; *,DODU.FLG            5743
11169 040012  062706  000020                    ADD     #20,SP          ;                       5732
11170 040016  000434                            BR      8$              ;                       5730
11171 040020  132777  000200  153176    7$:     BITB    #200,@ML.REG+50 ;                       5747
11172 040026  001030                            BNE     8$
11173 040030  104455                            TRAP    55              ;                       5750
11174 040032  000056                            .WORD   56
11175 040034  010464                            .WORD   ASYNC
11176 040036  024060                            .WORD   DUMPER
11177 040040  012746  007504                    MOV     #PHR.6,-(SP)    ;                       5751
11178 040044  012746  006276                    MOV     #WRD.1,-(SP)
11179 040050  012746  006754                    MOV     #WRD.43,-(SP)
```

```
11181                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
11182                                   ;                                                       14-May-1981 13:05:08    PA:<
11183
11184 040054  012746  007366                    MOV     #PHR.1,-(SP)
11185 040060  012746  006302                    MOV     #WRD.2,-(SP)
11186 040064  012746  006062                    MOV     #FIV.FMT,-(SP)
11187 040070  012746  000006                    MOV     #6,-(SP)
11188 040074  010600                           MOV     SP,R0                   ; SP,*
11189 040076  104414                           TRAP    14
11190 040100  012702  000001                    MOV     #1,R2                   ; *,DODU.FLG                                    5752
11191 040104  062706  000016                    ADD     #16,SP                  ;                                               5749
11192 040110  132777  000001  153116  8$:      BITB    #1,@ML.REG+60           ;                                               5755
11193 040116  001430                           BEQ     9$
11194 040120  104455                           TRAP    55                      ;                                               5758
11195 040122  000057                           .WORD   57
11196 040124  010464                           .WORD   ASYNC
11197 040126  024060                           .WORD   DUMPER
11198 040130  012746  006506                    MOV     #WRD.19,-(SP)           ;                                               5759
11199 040134  012746  010012                    MOV     #FNC.7,-(SP)
11200 040140  012746  007750                    MOV     #FNC.3,-(SP)
11201 040144  012746  006426                    MOV     #WRD.12,-(SP)
11202 040150  012746  007472                    MOV     #PHR.5,-(SP)
11203 040154  012746  006314                    MOV     #WRD.3,-(SP)
11204 040160  012746  006062                    MOV     #FIV.FMT,-(SP)
11205 040164  012746  000007                    MOV     #7,-(SP)
11206 040170  010600                           MOV     SP,R0                   ; SP,*
11207 040172  104414                           TRAP    14
11208 040174  062706  000020                    ADD     #20,SP                  ;                                               5757
11209 040200  032777  020000  153026  9$:      BIT     #20000,@ML.REG+60       ;                                               5762
11210 040206  001430                           BEQ     10$
11211 040210  104455                           TRAP    55                      ;                                               5765
11212 040212  000060                           .WORD   60
11213 040214  010464                           .WORD   ASYNC
11214 040216  024060                           .WORD   DUMPER
11215 040220  012746  006506                    MOV     #WRD.19,-(SP)           ;                                               5766
11216 040224  012746  010012                    MOV     #FNC.7,-(SP)
11217 040230  012746  007750                    MOV     #FNC.3,-(SP)
11218 040234  012746  006426                    MOV     #WRD.12,-(SP)
11219 040240  012746  007472                    MOV     #PHR.5,-(SP)
11220 040244  012746  006322                    MOV     #WRD.4,-(SP)
11221 040250  012746  006062                    MOV     #FIV.FMT,-(SP)
11222 040254  012746  000007                    MOV     #7,-(SP)
11223 040260  010600                           MOV     SP,R0                   ; SP,*
11224 040262  104414                           TRAP    14
11225 040264  062706  000020                    ADD     #20,SP                  ;                                               5764
11226 040270  005777  152740          10$:      TST     @ML.REG+60              ;                                               5769
11227 040274  001430                           BEQ     11$
11228 040276  104455                           TRAP    55                      ;                                               5772
11229 040300  000061                           .WORD   61
11230 040302  010464                           .WORD   ASYNC
11231 040304  024060                           .WORD   DUMPER
11232 040306  012746  010274                    MOV     #REG.3,-(SP)            ;                                               5773
11233 040312  012746  006436                    MOV     #WRD.13,-(SP)
11234 040316  012746  006442                    MOV     #WRD.14,-(SP)
11235 040322  012746  006506                    MOV     #WRD.19,-(SP)
```

```
11237                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
11238                                   ;                                                       14-May-1981 13:05:08    PA:<
11239
11240 040326  012746  010012                      MOV     #FNC.7,-(SP)
11241 040332  012746  007750                      MOV     #FNC.3,-(SP)
11242 040336  012746  006100                      MOV     #SIX.FMT,-(SP)
11243 040342  012746  000007                      MOV     #7,-(SP)
11244 040346  010600                              MOV     SP,R0                           ; SP,*
11245 040350  104414                              TRAP    14
11246 040352  062706  000020                      ADD     #20,SP                          ;                       5771
11247 040356  104467                    11$:      TRAP    67                              ;                       5774
11248 040360  006000                              ROR     R0
11249 040362  103002                              BHIS    12$
11250 040364  000167  177152                      JMP     1$
11251 040370  005302                    12$:      DEC     R2                              ; DODU.FLG              5778
11252 040372  001004                              BNE     13$
11253 040374  016700  153200                      MOV     ML.LUN,R0                       ;                       5781
11254 040400  104451                              TRAP    51
11255 040402  104444                              TRAP    44
11256 040404  005726                    13$:      TST     (SP)+                           ;                       5700
11257 040406  000207                              RTS     PC
11258
11259                                   ; Routine Size:  214 words
11260                                   ; Maximum stack depth per invocation:  19 words
11265
11266
11270
11271                                           .SBTTL  T16
11275 040410                            T16::
11276 040410  004767  177120            1$:     JSR     PC,$T16                         ;                       5783
11277 040414  104466                              TRAP    66
11278 040416  006000                              ROR     R0
11279 040420  103773                              BLO     1$
11280 040422  000207                              RTS     PC
11281
11282                                   ; Routine Size:  6 words
11283                                   ; Maximum stack depth per invocation:  0 words
11288
11289
11290 ;          5786  !<BLF/PAGE>
```

```
11292 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
11293 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (67)
11294
11295 ;          5787  !
11296 ;          5788  BGNTST
11297 ;          5789
11298 ;          5790  !++
11299 ;          5791  ! TEST NUMBER:  TST 17
11300 ;          5792  !
11301 ;          5793  ! TEST NAME: DIAGNOSTIC REGISTER TESTS
11302 ;          5794  !
11303 ;          5795  ! TEST DESCRIPTION:
11304 ;          5796  !       TEST THE DATA DIAG REGISTERS MLD1, MLD2, MLE2
11305 ;          5797  !       FOR 1'S/0'S, SHIFTING 1'S/0'S AND INITIALIZATION
11306 ;          5798  !
11307 ;          5799  !--
11308 ;          5800
11309 ;          5801  local
11310 ;          5802  CLR_DATA ,                                  !CLEAR DATA FOR INIT TEST
11311 ;          5803  SAVE ,                                      !TEMPORARY SAVE LOCATION
11312 ;          5804  ERR_FLG ,                                   !ERROR FLAG
11313 ;          5805  TST_PAT ,                                   !TEST PATTERN
11314 ;          5806  index ,                                     !POINTS TO REGISTER PRESENTLY BEING TESTED
11315 ;          5807  DODU_FLG;                                   !DROP UNIT FLAG
11316 ;          5808  DODU_FLG = ZERO;
11317 ;          5809  !
11318 ;          5810  !
11319 ;          5811  !       FIRST TEST THE REGISTERS FOR ONES AND ZEROES
11320 ;          5812  !
11321 ;          5813  !
11322 ;          5814  TST_PAT = ONES;                             !LOAD TEST PAT WITH ONES
11323 ;          5815
11324 ;          5816  incr TWICE from 0 to 1 do                  !REPEAT LOOP TWICE
11325 ;          5817      begin
11326 ;          5818
11327 ;          5819      incr REG_SEL from 11 to 13 do           !TEST ELEVEN WRITE/READ REGISTERS
11328 ;          5820          begin
11329 ;          5821          BGNSUB;
11330 ;          5822          CLR_MBUS;
11331 ;          5823          WRT_REG (.TST_PAT, .REG_SEL, index);    !WRITE TO THE REGISTER
11332 ;          5824          RD_REG (.TST_PAT, .REG_SEL, ERR_FLG);   !READ THE REGISTER
11333 ;          5825
11334 ;          5826          if .ERR_FLG IS_SET                      !SEE IF READ FOUND AN ERROR
11335 ;          5827          then
11336 ;          5828              begin                               !IF ERROR FLAG IS_SET THEN ERROR AND SET DODU_FLG
11337 ;          5829              ERRDF (121, ARR_DAT, DUMPER);       !ARRAY DATA MODULE FAILURE
11338 ;          5830              PRINTB (SIX_FMT, PHR_4, WRD_12, FNC_5, FNC_6, WRD_52, WRD_56);
11339 ;          5831              PRINTB (FMT_16, .ML_REG [.index, REGISTER_ADD], .WT_DATA, .RD_DATA);
11340 ;          5832              DODU_FLG = ONE;
11341 ;          5833              end;
11342 ;          5834
11343 ;          5835          ENDSUB;
11344 ;          5836          end;
11345 ;          5837
11346 ;          5838      TST_PAT = not .TST_PAT;                 !REPEAT AGAIN WITH COMPLIMENT DATA
11347 ;ML4                                                         14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
11348 ;                                                           14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (67)
```

```
11349
11350 ;          5839      end;
11351 ;          5840
11352 ;          5841  !
11353 ;          5842  !
11354 ;          5843  !       TEST THE REGISTERS FOR SHIFTING ONES AND ZEROES
11355 ;          5844  !
11356 ;          5845  !
11357 ;          5846  TST_PAT = ONE;                          !LOAD TST_PAT WITH A 1 IN A FILED OF 0'S
11358 ;          5847
11359 ;          5848  incr SHIFT from 0 to 15 do              !DO SHIFT 16 TIMES
11360 ;          5849      begin
11361 ;          5850
11362 ;          5851      incr TWICE from 0 to 1 do           !REPEAT LOOP TWICE
11363 ;          5852          begin
11364 ;          5853
11365 ;          5854          incr REG_SEL from 11 to 13 do   !TEST ELEVEN READ/WRITE REGISTERS
11366 ;          5855              begin
11367 ;          5856              BGNSUB;
11368 ;          5857              CLR_MBUS;
11369 ;          5858              WRT_REG (.TST_PAT, .REG_SEL, index);      !WRITE TO THE REGISTER
11370 ;          5859              RD_REG (.TST_PAT, .REG_SEL, ERR_FLG);     !READ THE REGISTER
11371 ;          5860
11372 ;          5861              if .ERR_FLG IS_SET                !SEE IF THE READ FOUND AN ERROR
11373 ;          5862              then
11374 ;          5863                  begin                         !IF THE ERROR FLAG IS SET THEN ERROR
11375 ;          5864                  ERRDF (122, ARR_DAT, DUMPER);  !ARRAY DATA MODULE FAILURE
11376 ;          5865                  PRINTB (SIX_FMT, PHR_4, WRD_12, FNC_5, FNC_6, WRD_52, WRD_56);
11377 ;          5866                  PRINTB (FMT_16, .ML_REG [.index, REGISTER_ADD], .WT_DATA, .RD_DATA);
11378 ;          5867                  DODU_FLG = ONE;
11379 ;          5868                  end;
11380 ;          5869
11381 ;          5870              ENDSUB;
11382 ;          5871              end;
11383 ;          5872
11384 ;          5873          TST_PAT = not .TST_PAT;           !REPEAT WITH A 0 IN A FIELD OF 1'S
11385 ;          5874          end;
11386 ;          5875
11387 ;          5876      TST_PAT = .TST_PAT^ONE;               !SHIFT THE 1 IN THE FIELD OF 0'S
11388 ;          5877      end;
11389 ;          5878
11390 ;          5879  !
11391 ;          5880  !
11392 ;          5881  !       NOW TEST THE REGISTERS FOR INITIALIZATION
11393 ;          5882  !
11394 ;          5883  !
11395 ;          5884  REG_INIT_FLG = ONE;
11396 ;          5885  TST_PAT = ONES;                          !BACKGROUND PATTERN
11397 ;          5886
11398 ;          5887  incr TWICE from 0 to 1 do               !REPEAT LOOP TWICE
11399 ;          5888      begin
11400 ;          5889
11401 ;          5890      incr REG_SEL from 11 to 13 do       !TEST THIRTEEN REGISTERS
11402 ;ML4                                                     14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
11403 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (67)
11404
11405 ;          5891          begin
```

```
 11406 ;          5892          BGNSUB;
 11407 ;          5893          CLR_MBUS;
 11408 ;          5894          WRT_REG (.TST_PAT, .REG_SEL, index);    !WRITE REGISTER WITH BACKGROUND
 11409 ;          5895          CLR_DATA = (.HI) or (.IGNORE);          !CALCULATE THE CLEARED DATA PATTERN
 11410 ;          5896          REG_INIT_FLG = ONE;
 11411 ;          5897          RD_REG (.CLR_DATA, .REG_SEL, ERR_FLG);  !READ THE REGISTER FOR THE CLEARED DATA PAT
 11412 ;          5898
 11413 ;          5899          if .ERR_FLG IS_SET                      !SEE IF READ FOUND AN ERROR
 11414 ;          5900          then
 11415 ;          5901              begin                               !IF ERROR FLAG IS_SET THEN ERROR AND SET DODU_FLG
 11416 ;          5902              ERRDF (118, ARR_DAT, DUMPER);       !ARRAY DATA MODULE
 11417 ;          5903              PRINTB (FIV_FMT, PHR_4, WRD_12, WRD_52, FNC_23, WRD_56);
 11418 ;          5904              PRINTB (FMT_16, .ML_REG [.index, REGISTER_ADD], .CLR_DATA, .RD_DATA);
 11419 ;          5905              DODU_FLG = ONE;
 11420 ;          5906              end;
 11421 ;          5907
 11422 ;          5908          ENDSUB;
 11423 ;          5909          end;
 11424 ;          5910
 11425 ;          5911      TST_PAT = not .TST_PAT;                     !REPEAT WITH COMPLIMENT BACKGROUND PAT
 11426 ;          5912      end;
 11427 ;          5913
 11428 ;          5914  REG_INIT_FLG = ZERO;                           !CLEAR THE FLAG
 11429 ;          5915
 11430 ;          5916  if .DODU_FLG IS_SET                           !DROP THIS UNIT IF DODU_FLG SET
 11431 ;          5917  then
 11432 ;          5918      begin
 11433 ;          5919      DODU (.ML_LUN);
 11434 ;          5920      DOCLN;
 11435 ;          5921      end;
 11436 ;          5922
 11437 ;          5923  ENDTST;
 11441
 11442                                    .SBTTL  $T17
 11446 040424  004167  143466      $T17:  JSR     R1,$SAVE5          ;                                  5785
 11447 040430  024646                     CMP     -(SP),-(SP)
 11448 040432  005046                     CLR     -(SP)              ; DODU.FLG                         5808
 11449 040434  012703  177777             MOV     #-1,R3             ; *,TST.PAT                        5814
 11450 040440  005001                     CLR     R1                 ; TWICE                            5816
 11451 040442  012702  000013       1$:   MOV     #13,R2             ; *,REG.SEL                        5819
 11452 040446  104402               2$:   TRAP    2                  ;                                  5820
 11453 040450  152777  000040  152536     BISB    #40,@ML.REG+40     ;                                  5821
 11454 040456  016705  153120             MOV     ML.DUT,R5
 11455 040462  042705  177770             BIC     #177770,R5
 11456 040466  142777  000007  152520     BICB    #7,@ML.REG+40
 11457                               ;ML4                                    14-May-1981 13:48:54    TOPS
 11458                               ;                                       14-May-1981 13:05:08    PA:<
 11459
 11460 040474  150577  152514             BISB    R5,@ML.REG+40
 11461 040500  010346                     MOV     R3,-(SP)           ; TST.PAT,*                        5823
 11462 040502  010246                     MOV     R2,-(SP)           ; REG.SEL,*
 11463 040504  012746  000012             MOV     #12,-(SP)
 11464 040510  060616                     ADD     SP,(SP)            ; INDEX,*
 11465 040512  004767  162364             JSR     PC,WRT.REG
 11466 040516  010316                     MOV     R3,(SP)            ; TST.PAT,*                        5824
 11467 040520  010246                     MOV     R2,-(SP)           ; REG.SEL,*
 11468 040522  012746  000014             MOV     #14,-(SP)
```

```
11469 040526  060616                        ADD     SP,(SP)                 ; ERR.FLG,*
11470 040530  004767  162734                JSR     PC,RD.REG
11471 040534  026627  000014  000001        CMP     14(SP),#1               ; ERR.FLG,*                     5826
11472 040542  001054                        BNE     3$
11473 040544  104455                        TRAP    55                      ;                               5829
11474 040546  000171                        .WORD   171
11475 040550  010570                        .WORD   ARR.DAT
11476 040552  024060                        .WORD   DUMPER
11477 040554  012746  007122                MOV     #WRD.56,-(SP)           ;                               5830
11478 040560  012746  007066                MOV     #WRD.52,-(SP)
11479 040564  012746  010004                MOV     #FNC.6,-(SP)
11480 040570  012746  007774                MOV     #FNC.5,-(SP)
11481 040574  012746  006426                MOV     #WRD.12,-(SP)
11482 040600  012746  007454                MOV     #PHR.4,-(SP)
11483 040604  012746  006100                MOV     #SIX.FMT,-(SP)
11484 040610  012746  000007                MOV     #7,-(SP)
11485 040614  010600                        MOV     SP,R0                   ; SP,*
11486 040616  104414                        TRAP    14
11487 040620  016716  152304                MOV     RD.DATA,(SP)            ;                               5831
11488 040624  016746  152276                MOV     WT.DATA,-(SP)
11489 040630  016600  000040                MOV     40(SP),R0               ; INDEX,*
11490 040634  006300                        ASL     R0
11491 040636  006300                        ASL     R0
11492 040640  006300                        ASL     R0
11493 040642  016046  013154                MOV     ML.REG(R0),-(SP)
11494 040646  012746  005216                MOV     #FMT.16,-(SP)
11495 040652  012746  000004                MOV     #4,-(SP)
11496 040656  010600                        MOV     SP,R0                   ; SP,*
11497 040660  104414                        TRAP    14
11498 040662  012766  000001  000042        MOV     #1,42(SP)               ; *,DODU.FLG                    5832
11499 040670  062706  000030                ADD     #30,SP                  ;                               5828
11500 040674  062706  000012          3$:   ADD     #12,SP                  ;                               5820
11501 040700  104467                        TRAP    67                      ;                               5833
11502 040702  006000                        ROR     R0
11503 040704  103660                        BLO     2$
11504 040706  005202                        INC     R2                      ; REG.SEL                       5819
11505 040710  020227  000015                CMP     R2,#15                  ; REG.SEL,*
11506 040714  003654                        BLE     2$
11507 040716  005103                        COM     R3                      ; TST.PAT                       5838
11508 040720  005201                        INC     R1                      ; TWICE                         5816
11509 040722  020127  000001                CMP     R1,#1                   ; TWICE,*
11510 040726  003645                        BLE     1$
11511 040730  012703  000001                MOV     #1,R3                   ; *,TST.PAT                     5846
11512                             ;ML4                                                  14-May-1981 13:48:54    TOPS
11513                             ;                                                     14-May-1981 13:05:08    PA:<
11514
11515 040734  005004                  CLR     R4                      ; SHIFT                         5848
11516 040736  005001            4$:   CLR     R1                      ; TWICE                         5851
11517 040740  012702  000013      5$:   MOV     #13,R2                  ; *,REG.SEL                     5854
11518 040744  104402            6$:   TRAP    2                       ;                               5855
11519 040746  152777  000040  152240  BISB    #40,@ML.REG+40          ;                               5856
11520 040754  016705  152622          MOV     ML.DUT,R5
11521 040760  042705  177770          BIC     #177770,R5
11522 040764  142777  000007  152222  BICB    #7,@ML.REG+40
11523 040772  150577  152216          BISB    R5,@ML.REG+40
11524 040776  010346                  MOV     R3,-(SP)                ; TST.PAT,*                     5858
11525 041000  010246                  MOV     R2,-(SP)                ; REG.SEL,*
```

```
11526 041002  012746  000012                 MOV     #12,-(SP)
11527 041006  060616                         ADD     SP,(SP)                 ; INDEX,*
11528 041010  004767  162066                 JSR     PC,WRT.REG
11529 041014  010316                         MOV     R3,(SP)                 ; TST.PAT,*                     5859
11530 041016  010246                         MOV     R2,-(SP)                ; REG.SEL,*
11531 041020  012746  000014                 MOV     #14,-(SP)
11532 041024  060616                         ADD     SP,(SP)                 ; ERR.FLG,*
11533 041026  004767  162436                 JSR     PC,RD.REG
11534 041032  026627  000014  000001         CMP     14(SP),#1               ; ERR.FLG,*                     5861
11535 041040  001054                         BNE     7$
11536 041042  104455                         TRAP    55                      ;                               5864
11537 041044  000172                         .WORD   172
11538 041046  010570                         .WORD   ARR.DAT
11539 041050  024060                         .WORD   DUMPER
11540 041052  012746  007122                 MOV     #WRD.56,-(SP)           ;                               5865
11541 041056  012746  007066                 MOV     #WRD.52,-(SP)
11542 041062  012746  010004                 MOV     #FNC.6,-(SP)
11543 041066  012746  007774                 MOV     #FNC.5,-(SP)
11544 041072  012746  006426                 MOV     #WRD.12,-(SP)
11545 041076  012746  007454                 MOV     #PHR.4,-(SP)
11546 041102  012746  006100                 MOV     #SIX.FMT,-(SP)
11547 041106  012746  000007                 MOV     #7,-(SP)
11548 041112  010600                         MOV     SP,R0                   ; SP,*
11549 041114  104414                         TRAP    14
11550 041116  016716  152006                 MOV     RD.DATA,(SP)            ;                               5866
11551 041122  016746  152000                 MOV     WT.DATA,-(SP)
11552 041126  016600  000040                 MOV     40(SP),R0               ; INDEX,*
11553 041132  006300                         ASL     R0
11554 041134  006300                         ASL     R0
11555 041136  006300                         ASL     R0
11556 041140  016046  013154                 MOV     ML.REG(R0),-(SP)
11557 041144  012746  005216                 MOV     #FMT.16,-(SP)
11558 041150  012746  000004                 MOV     #4,-(SP)
11559 041154  010600                         MOV     SP,R0                   ; SP,*
11560 041156  104414                         TRAP    14
11561 041160  012766  000001  000042         MOV     #1,42(SP)               ; *,DODU.FLG                    5867
11562 041166  062706  000030                 ADD     #30,SP                  ;                               5863
11563 041172  062706  000012           7$:   ADD     #12,SP                  ;                               5855
11564 041176  104467                         TRAP    67                      ;                               5868
11565 041200  006000                         ROR     R0
11566 041202  103660                         BLO     6$
11567                               ;ML4                                                    14-May-1981 13:48:54    TOPS
11568                               ;                                                       14-May-1981 13:05:08    PA:<
11569
11570 041204  005202                         INC     R2                      ; REG.SEL                       5854
11571 041206  020227  000015                 CMP     R2,#15                  ; REG.SEL,*
11572 041212  003654                         BLE     6$
11573 041214  005103                         COM     R3                      ; TST.PAT                       5873
11574 041216  005201                         INC     R1                      ; TWICE                         5851
11575 041220  020127  000001                 CMP     R1,#1                   ; TWICE,*
11576 041224  003645                         BLE     5$
11577 041226  006303                         ASL     R3                      ; TST.PAT                       5876
11578 041230  005204                         INC     R4                      ; SHIFT                         5848
11579 041232  020427  000017                 CMP     R4,#17                  ; SHIFT,*
11580 041236  003637                         BLE     4$
11581 041240  012767  000001  151670         MOV     #1,REG.INIT.FLG         ;                               5884
11582 041246  012703  177777                 MOV     #-1,R3                  ; *,TST.PAT                     5885
```

```
11583 041252  005004                        CLR     R4                      ; TWICE                         5887
11584 041254  012702  000013          8$:   MOV     #13,R2                  ; *,REG.SEL                     5890
11585 041260  104402                9$:   TRAP    2                       ;                               5891
11586 041262  152777  000040  151724        BISB    #40,@ML.REG+40          ;                               5892
11587 041270  016701  152306                MOV     ML.DUT,R1
11588 041274  042701  177770                BIC     #177770,R1
11589 041300  142777  000007  151706        BICB    #7,@ML.REG+40
11590 041306  150177  151702                BISB    R1,@ML.REG+40
11591 041312  010346                        MOV     R3,-(SP)                ; TST.PAT,*                     5894
11592 041314  010246                        MOV     R2,-(SP)                ; REG.SEL,*
11593 041316  012746  000012                MOV     #12,-(SP)
11594 041322  060616                        ADD     SP,(SP)                 ; INDEX,*
11595 041324  004767  161552                JSR     PC,WRT.REG
11596 041330  016600  000012                MOV     12(SP),R0               ; INDEX,*                       5895
11597 041334  006300                        ASL     R0
11598 041336  006300                        ASL     R0
11599 041340  006300                        ASL     R0
11600 041342  010001                        MOV     R0,R1
11601 041344  016105  013156                MOV     ML.REG+2(R1),R5         ; *,CLR.DATA
11602 041350  056105  013162                BIS     ML.REG+6(R1),R5         ; *,CLR.DATA
11603 041354  012767  000001  151554        MOV     #1,REG.INIT.FLG         ;                               5896
11604 041362  010516                        MOV     R5,(SP)                 ; CLR.DATA,*                    5897
11605 041366  010246                        MOV     R2,-(SP)                ; REG.SEL,*
11606 041366  012746  000014                MOV     #14,-(SP)
11607 041372  060616                        ADD     SP,(SP)                 ; ERR.FLG,*
11608 041374  004767  162070                JSR     PC,RD.REG
11609 041400  026627  000014  000001        CMP     14(SP),#1               ; ERR.FLG,*                     5899
11610 041406  001044                        BNE     10$
11611 041410  104455                        TRAP    55                      ;                               5902
11612 041412  000166                        .WORD   166
11613 041414  010570                        .WORD   ARR.DAT
11614 041416  024060                        .WORD   DUMPER
11615 041420  012746  007122                MOV     #WRD.56,-(SP)           ;                               5903
11616 041424  012746  010242                MOV     #FNC.23,-(SP)
11617 041430  012746  007066                MOV     #WRD.52,-(SP)
11618 041434  012746  006426                MOV     #WRD.12,-(SP)
11619 041440  012746  007454                MOV     #PHR.4,-(SP)
11620 041444  012746  006062                MOV     #FIV.FMT,-(SP)
11621 041450  012746  000006                MOV     #6,-(SP)
11622                               ;ML4                                                14-May-1981 13:48:54    TOPS
11623                               :                                                   14-May-1981 13:05:08    PA:<
11624
11625 041454  010600                        MOV     SP,R0                   ; SP,*
11626 041456  104414                        TRAP    14
11627 041460  016716  151444                MOV     RD.DATA,(SP)            ;                               5904
11628 041464  010546                        MOV     R5,-(SP)                ; CLR.DATA,*
11629 041466  016146  013154                MOV     ML.REG(R1),-(SP)
11630 041472  012746  005216                MOV     #FMT.16,-(SP)
11631 041476  012746  000004                MOV     #4,-(SP)
11632 041502  010600                        MOV     SP,R0                   ; SP,*
11633 041504  104414                        TRAP    14
11634 041506  012766  000001  000040        MOV     #1,40(SP)               ; *,DODU.FLG                    5905
11635 041514  062706  000026                ADD     #26,SP                  ;                               5901
11636 041520  062706  000012          10$:  ADD     #12,SP                  ;                               5891
11637 041524  104467                        TRAP    67                      ;                               5906
11638 041526  006000                        ROR     R0
11639 041530  103653                        BLO     9$
```

```
11640 041532  005202                    INC     R2                      ; REG.SEL                       5890
11641 041534  020227  000015            CMP     R2,#15                  ; REG.SEL,*
11642 041540  003647                    BLE     9$
11643 041542  005103                    COM     R3                      ; TST.PAT                       5911
11644 041544  005204                    INC     R4                      ; TWICE                         5887
11645 041546  020427  000001            CMP     R4,#1                   ; TWICE,*
11646 041552  003640                    BLE     8$
11647 041554  005067  151356            CLR     REG.INIT.FLG            ;                               5914
11648 041560  021627  000001            CMP     (SP),#1                 ; DODU.FLG,*                    5916
11649 041564  001004                    BNE     11$
11650 041566  016700  152006            MOV     ML.LUN,R0               ;                               5919
11651 041572  104451                    TRAP    51
11652 041574  104444                    TRAP    44
11653 041576  062706  000006    11$:    ADD     #6,SP                   ;                               5785
11654 041602  000207                    RTS     PC
11655
11656                           ; Routine Size:  312 words
11657                           ; Maximum stack depth per invocation:  26 words
11662
11663
11667
11668                                   .SBTTL  T17
11672 041604                    T17::
11673 041604  004767  176614    1$:     JSR     PC,$T17                 ;                               5921
11674 041610  104466                    TRAP    66
11675 041612  006000                    ROR     R0
11676 041614  103773                    BLO     1$
11677                           ;ML4                                            14-May-1981 13:48:54    TOPS
11678                           ;                                               14-May-1981 13:05:08    PA:<
11679
11680 041616  000207                    RTS     PC
11681
11682                           ; Routine Size:  6 words
11683                           ; Maximum stack depth per invocation:  0 words
11688
11689
11690 ;          5924   !<BLF/PAGE>
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (68)

```
11692 ;ML4
11693 ;
11694
11695 ;          5925  !
11696 ;          5926  !
11697 ;          5927  BGNTST;
11698 ;          5928
11699 ;          5929  !++
11700 ;          5930  ! TEST NUMBER:  TST 18
11701 ;          5931  !
11702 ;          5932  ! TEST NAME:    COMPOSIT ERROR TEST
11703 ;          5933  !
11704 ;          5934  ! TEST DESCRIPTION:
11705 ;          5935  !       TEST TO SEE IF SETTING EACH
11706 ;          5936  !       BIT IN THE ERROR REGISTER
11707 ;          5937  !       CAUSES A COMPOSIT ERROR BY:
11708 ;          5938  !
11709 ;          5939  !       WRITING A SHIFTING ONE THROUGH
11710 ;          5940  !       THE ERROR REGISTER (SKIPPING THE
11711 ;          5941  !       READ ONLY BITS) AND TESTING THE
11712 ;          5942  !       COMPOSIT ERROR BIT IN MLDS
11713 ;          5943  !       FOR BEING SET AFTER EACH
11714 ;          5944  !       WRITE.
11715 ;          5945  !
11716 ;          5946  !--
11717 ;          5947
11718 ;          5948  local
11719 ;          5949      DODU_FLG,                          !DROP UNIT FLAG
11720 ;          5950      DAT_PAT,                           !DATA PATTERN
11721 ;          5951      SKIP_MASK;                         !POINTS TO MLER READ ONLY BITS
11722 ;          5952
11723 ;          5953  CLR_MBUS;
11724 ;          5954  DODU_FLG = ZERO;
11725 ;          5955  SKIP_MASK = %o'163157';                !LOAD SKIP MASK
11726 ;          5956  DAT_PAT = ONE;                         !DATA PATTERN SET BIT 0 IN MLER
11727 ;          5957  DODU_FLG = ZERO;
11728 ;          5958
11729 ;          5959  incr COUNT from 0 to 15 do             !WRITE AND SHIFT DATA PAT TO MLER 16 TIMES
11730 ;          5960      begin
11731 ;          5961
11732 ;          5962      if (.DAT_PAT and .SKIP_MASK) neq ZERO   !SKIP IF DAT_PAT FALLS ON READ ONLY BIT
11733 ;          5963      then
11734 ;          5964          begin
11735 ;          5965          BGNSUB;
11736 ;          5966          MLER = .DAT_PAT;               !WRITE DATA_PAT TO MLER
11737 ;          5967
11738 ;          5968          if .COMP_ERR IS_NOT_SET        !SEE IF DAT_PAT CAUSED A COMP ERROR
11739 ;          5969          then
11740 ;          5970              begin
11741 ;          5971              ERRDF (50, ASYNC, DUMPER);     !ERROR IF NO COMP ERROR
11742 ;          5972              PRINTB (FOR_FMT, FNC_8, PHR_1, WRD_12, FNC_8);
11743 ;          5973              PRINTB (FMT_4, .DAT_PAT);
11744 ;          5974              DODU_FLG = ONE;
11745 ;          5975              end;
11746 ;          5976
```

```
11748 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
11749 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (68)
11750
11751 ;          5977          ENDSUB;
11752 ;          5978          end;
11753 ;          5979
11754 ;          5980      DAT_PAT = .DAT_PAT^ONE;                !SHIFT DAT_PAT TO NEXT BIT AND REPEAT
11755 ;          5981      end;
11756 ;          5982
11757 ;          5983  if .DODU_FLG IS_SET                        !DROP UNIT IF DODU_FLG IS_SET
11758 ;          5984  then
11759 ;          5985      begin
11760 ;          5986      DODU (.ML_LUN);
11761 ;          5987      DOCLN;
11762 ;          5988      end;
11763 ;          5989
11764 ;          5990  ENDTST;
11768
11769                                        .SBTTL  $T18
11773 041620  004167  142252          $T18: JSR     R1,$SAVE4                ;                              5923
11774 041624  152777  000040  151362        BISB    #40,@ML.REG+40           ;                              5951
11775 041632  016704  151744              MOV     ML.DUT,R4
11776 041636  042704  177770              BIC     #177770,R4
11777 041642  142777  000007  151344        BICB    #7,@ML.REG+40
11778 041650  150477  151340              BISB    R4,@ML.REG+40
11779 041654  005001                      CLR     R1                       ; DODU.FLG                     5954
11780 041656  012704  163157              MOV     #-14621,R4               ; *,SKIP.MASK                  5955
11781 041662  012702  000001              MOV     #1,R2                    ; *,DAT.PAT                    5956
11782 041666  005003                      CLR     R3                       ; COUNT                        5959
11783 041670  030204                1$:   BIT     R2,R4                    ; DAT.PAT,SKIP.MASK            5962
11784 041672  001447                      BEQ     4$
11785 041674  104402                2$:   TRAP    2                        ;                              5964
11786 041676  010277  151332              MOV     R2,@ML.REG+60            ; DAT.PAT,*                    5966
11787 041702  032777  040000  151314        BIT     #40000,@ML.REG+50        ;                              5968
11788 041710  001035                      BNE     3$
11789 041712  104455                      TRAP    55                       ;                              5971
11790 041714  000062                      .WORD   62
11791 041716  010464                      .WORD   ASYNC
11792 041720  024060                      .WORD   DUMPER
11793 041722  012746  010022              MOV     #FNC.8,-(SP)             ;                              5972
11794 041726  012746  006426              MOV     #WRD.12,-(SP)
11795 041732  012746  007366              MOV     #PHR.1,-(SP)
11796 041736  012746  010022              MOV     #FNC.8,-(SP)
11797 041742  012746  006046              MOV     #FOR.FMT,-(SP)
11798 041746  012746  000005              MOV     #5,-(SP)
11799 041752  010600                      MOV     SP,R0                    ; SP,*
11800 041754  104414                      TRAP    14
11801 041756  010216                      MOV     R2,(SP)                  ; DAT.PAT,*                    5973
11802 041760  012746  004400              MOV     #FMT.4,-(SP)
```

```
                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                        ;                                                       14-May-1981 13:05:08    PA:<
                        
041764  012746  000002          MOV     #2,-(SP)
041770  010600                  MOV     SP,R0                   ; SP,*
041772  104414                  TRAP    14
041774  012701  000001          MOV     #1,R1                   ; *,DODU.FLG                            5974
042000  062706  000020          ADD     #20,SP                  ;                                       5970
042004  104467          3$:     TRAP    67                      ;                                       5975
042006  006000                  ROR     R0
042010  103731                  BLO     2$
042012  006302          4$:     ASL     R2                      ; DAT.PAT                               5980
042014  005203                  INC     R3                      ; COUNT                                 5959
042016  020327  000017          CMP     R3,#17                  ; COUNT,*
042022  003722                  BLE     1$
042024  005301                  DEC     R1                      ; DODU.FLG                              5983
042026  001004                  BNE     5$
042030  016700  151544          MOV     ML.LUN,R0               ;                                       5986
042034  104451                  TRAP    51
042036  104444                  TRAP    44
042040  000207          5$:     RTS     PC                      ;                                       5923
                        
                        ; Routine Size:  73 words
                        ; Maximum stack depth per invocation:  13 words
                        
                        
                        
                                .SBTTL  T18
042042                  T18::
042042  004767  177552  1$:     JSR     PC,$T18                 ;                                       5988
042046  104466                  TRAP    66
042050  006000                  ROR     R0
042052  103773                  BLO     1$
042054  000207                  RTS     PC
                        
                        ; Routine Size:  6 words
                        ; Maximum stack depth per invocation:  0 words
                        
                        
                        
;               5991    !<BLF/PAGE>
```

```
11859 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
11860 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (69)
11861
11862 ;          5992  !
11863 ;          5993  !
11864 ;          5994  BGNTST;
11865 ;          5995
11866 ;          5996  !++
11867 ;          5997  ! TEST NUMBER:  TST 19
11868 ;          5998  !
11869 ;          5999  ! TEST NAME:    ATA BIT TEST
11870 ;          6000  !
11871 ;          6001  ! TEST DESCRIPTION:
11872 ;          6002  !
11873 ;          6003  !       TEST THE ATA BIT FOR SETTING
11874 ;          6004  !       BY:
11875 ;          6005  !
11876 ;          6006  !       1.  SETTING A BIT IN THE ERROR
11877 ;          6007  !           REGISTER.
11878 ;          6008  !
11879 ;          6009  !           TEST THE ATA BIT FOR CLEARING
11880 ;          6010  !           AFTER BEING SET BY:
11881 ;          6011  !           1.  WRITING A FUNCTION TO MLCS1.
11882 ;          6012  !           2.  WRITING A ONE INTO THIS
11883 ;          6013  !               UNITS ATA BIT
11884 ;          6014  !
11885 ;          6015  !       TEST THE ATA BIT FOR NOT CLEARING
11886 ;          6016  !       AFTER BEING SET BY:
11887 ;          6017  !
11888 ;          6018  !       1.  WRITING A ONE INTO THE
11889 ;          6019  !           ATA BIT OF THE OTHER
11890 ;          6020  !           UNITS.
11891 ;          6021  !
11892 ;          6022  !--
11893 ;          6023
11894 ;          6024  local
11895 ;          6025      ATA_SAVE : bitvector [8],          !STORES ALL 8 ATA BITS ON READS AND WRITES
11896 ;          6026      DAT_PAT;                           !DATA PATTERN
11897 ;          6027
11898 ;          6028  CLR_MBUS;
11899 ;          6029  MLER = ONE;                            !SET THE ATA BIT
11900 ;          6030  MLER = ZERO;
11901 ;          6031  ATA_SAVE = .MLAS;                      !READ THE ATTN REGISTER
11902 ;          6032
11903 ;          6033  if .ATA_SAVE [.ML_DUT] IS_NOT_SET      !SEE IF THIS DRIVES ATA BIT IS SET
11904 ;          6034  then
11905 ;          6035      begin
11906 ;          6036      ERRDF (51, ASYNC, DUMPER);         !ERROR AND EXIT_TST IF NOT SET
11907 ;          6037      PRINTB (FOR_FMT, WRD_15, PHR_1, WRD_11, FNC_8);
11908 ;          6038      EXIT_TST;
11909 ;          6039      end;
11910 ;          6040
11911 ;          6041  if .ATTN IS_NOT_SET                    !SEE IF THE ATTN BIT IS SET
11912 ;          6042  then
11913 ;          6043      begin
```

```
11915 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
11916 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (69)
11917
11918 ;         6044      ERRDF (52, ASYNC, DUMPER);                  !ERROR AND EXIT_TST IF NOT SET
11919 ;         6045      PRINTB (FIV_FMT, REG_2, WRD_16, PHR_1, WRD_11, FNC_8);
11920 ;         6046      EXIT_TST;
11921 ;         6047      end;
11922 ;         6048
11923 ;         6049  MLCS1 = NOOP;                                  !TRY TO CLEAR THE ATA BIT WITH NOOP FUNC
11924 ;         6050
11925 ;         6051  if .ATTN IS_SET                               !SEE IF ATA GOT CLEARED
11926 ;         6052  then
11927 ;         6053      begin
11928 ;         6054      ERRDF (53, ASYNC, DUMPER);                !ERROR AND EXIT_TST IF SET
11929 ;         6055      PRINTB (FOR_FMT, WRD_15, PHR_2, WRD_11, FNC_2, WRD_19);
11930 ;         6056      EXIT_TST;
11931 ;         6057      end;
11932 ;         6058
11933 ;         6059  ATA_SAVE = .MLAS;                             !READ THE ATTENTION REGISTER
11934 ;         6060
11935 ;         6061  if .ATA_SAVE [.ML_DUT] IS_SET                 !SEE IF THE ATA REG GOT CLEARED BY NO-OP
11936 ;         6062  then
11937 ;         6063      begin
11938 ;         6064      ERRDF (58, ASYNC, DUMPER);
11939 ;         6065      PRINTB (FIV_FMT, WRD_15, PHR_2, WRD_11, FNC_2, WRD_19);
11940 ;         6066      end;
11941 ;         6067
11942 ;         6068  BGNSUB;
11943 ;         6069  MLER = ONE;                                   !SET THE ATA BIT
11944 ;         6070  MLER = ZERO;
11945 ;         6071  ATA_SAVE = ZEROES;                            !CLEAR ATA_SAVE
11946 ;         6072  ATA_SAVE [.ML_DUT] = ONE;                     !SET ATA_SAVE FOR THIS DRIVE
11947 ;         6073  MLAS = .ATA_SAVE;                             !TRY TO CLEAR THE ATA BY WRITING TO IT.
11948 ;         6074
11949 ;         6075  if .ATTN IS_SET                               !SEE IF THE ATA GOT CLEARED
11950 ;         6076  then
11951 ;         6077      begin
11952 ;         6078      ERRDF (54, ASYNC, DUMPER);                !ERROR IF NOT CLEARED
11953 ;         6079      PRINTB (FIV_FMT, WRD_15, PHR_2, WRD_11, WRD_17, REG_5);
11954 ;         6080      end;
11955 ;         6081
11956 ;         6082  ENDSUB;
11957 ;         6083  BGNSUB;
11958 ;         6084  MLER = ONE;                                   !SET THE ATA BIT
11959 ;         6085  MLER = ZERO;
11960 ;         6086  DAT_PAT = ONE;                                !DATA PATTERN OF ONE IN FIELD OF ZEROES
11961 ;         6087
11962 ;         6088  incr ATA_SEL from 0 to 7 do                   !REPEAT LOOP 8 TIMES
11963 ;         6089      begin
11964 ;         6090
11965 ;         6091      if .ATA_SEL neq .ML_DUT                   !SKIP IF ATA_SEL EQLS THIS DRIVE NO.
11966 ;         6092      then
11967 ;         6093          begin
11968 ;         6094          MLAS = .DAT_PAT;                      !WRITE DAT PAT TO ATA REGISTER
11969 ;         6095          ATA_SAVE = .MLAS;                     !READ ATA REG BACK
```

```
 11971 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
 11972 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (69)
 11973
 11974 ;          6096
 11975 ;          6097          if .ATA_SAVE [.ML_DUT] IS_NOT_SET       !SEE IF THIS DRIVE ATA IS CLEARED
 11976 ;          6098          then
 11977 ;          6099              begin
 11978 ;          6100              ERRDF (55, ASYNC, DUMPER);          !ERROR AND EXIT LOOP IF CLEARED
 11979 ;          6101              PRINTB (SIX_FMT, WRD_15, PHR_6, WRD_11, WRD_17, REG_5, PHR_7);
 11980 ;          6102              PRINTB (FMT_7, .DAT_PAT);
 11981 ;          6103              exitloop;
 11982 ;          6104              end;
 11983 ;          6105
 11984 ;          6106          end;
 11985 ;          6107
 11986 ;          6108      DAT_PAT = .DAT_PAT^ONE;                 !SHIFT DAT_PAT AND REPEAT
 11987 ;          6109      end;
 11988 ;          6110
 11989 ;          6111  ENDSUB;
 11990 ;          6112  ENDTST;
 11994
 11995                                        .SBTTL  $T19
 11999 042056  004167  141776          $T19:  JSR     R1,$SAVE3                ;                          5990
 12000 042062  152777  000040  151124        BISB    #40,@ML.REG+40           ;                          6026
 12001 042070  016703  151506                MOV     ML.DUT,R3
 12002 042074  042703  177770                BIC     #177770,R3
 12003 042100  142777  000007  151106        BICB    #7,@ML.REG+40
 12004 042106  150377  151102                BISB    R3,@ML.REG+40
 12005 042112  012777  000001  151114        MOV     #1,@ML.REG+60            ;                          6029
 12006 042120  005077  151110                CLR     @ML.REG+60               ;                          6030
 12007 042124  017746  151114                MOV     @ML.REG+70,-(SP.         ; *,ATA.SAVE               6031
 12008 042130  016701  151446                MOV     ML.DUT,R1                ;                          6033
 12009 042134  006201                        ASR     R1
 12010 042136  006201                        ASR     R1
 12011 042140  006201                        ASR     R1
 12012 042142  010600                        MOV     SP,R0                    ; ATA.SAVE,*
 12013 042144  060001                        ADD     R0,R1
 12014 042146  010146                        MOV     R1,-(SP)
 12015 042150  016746  151426                MOV     ML.DUT,-(SP)
 12016 042154  042716  177770                BIC     #177770,(SP)
 12017 042160  012746  000001                MOV     #1,-(SP)
 12018 042164  005046                        CLR     -(SP)
 12019 042166  004767  140746                JSR     PC,BL$GT2
 12020 042172  062706  000010                ADD     #10,SP
 12021 042176  005700                        TST     R0
 12022 042200  001026                        BNE     1$
 12023 042202  104455                        TRAP    55                       ;                          6036
 12024 042204  000063                        .WORD   63
 12025 042206  010464                        .WORD   ASYNC
```

```
12027                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
12028                                   ;                                                       14-May-1981 13:05:08    PA:<
12029
12030 042210  024060                            .WORD   DUMPER
12031 042212  012746  010022                    MOV     #FNC.8,-(SP)            ;                                       6037
12032 042216  012746  006416                    MOV     #WRD.11,-(SP)
12033 042222  012746  007366                    MOV     #PHR.1,-(SP)
12034 042226  012746  006454                    MOV     #WRD.15,-(SP)
12035 042232  012746  006046                    MOV     #FOR.FMT,-(SP)
12036 042236  012746  000005                    MOV     #5,-(SP)
12037 042242  010600                            MOV     SP,R0                   ; SP,*
12038 042244  104414                            TRAP    14
12039 042246  104463                            TRAP    63
12040 042250  062706  000014                    ADD     #14,SP                  ;                                       6033
12041 042254  000467                            BR      4$                      ;                                       6035
12042 042256  032777  100000  150740  1$:       BIT     #100000,@ML.REG+50      ;                                       6041
12043 042264  001026                            BNE     2$
12044 042266  104455                            TRAP    55                      ;                                       6044
12045 042270  000064                            .WORD   64
12046 042272  010464                            .WORD   ASYNC
12047 042274  024060                            .WORD   DUMPER
12048 042276  012746  010022                    MOV     #FNC.8,-(SP)            ;                                       6045
12049 042302  012746  006416                    MOV     #WRD.11,-(SP)
12050 042306  012746  007366                    MOV     #PHR.1,-(SP)
12051 042312  012746  006462                    MOV     #WRD.16,-(SP)
12052 042316  012746  010266                    MOV     #REG.2,-(SP)
12053 042322  012746  006062                    MOV     #FIV.FMT,-(SP)
12054 042326  012746  000006                    MOV     #6,-(SP)
12055 042332  010600                            MOV     SP,R0                   ; SP,*
12056 042334  104414                            TRAP    14
12057 042336  104463                            TRAP    63
12058 042340  000433                            BR      3$                      ;                                       6041
12059 042342  012777  000001  150604  2$:       MOV     #1,@ML.REG              ;                                       6049
12060 042350  005777  150650                    TST     @ML.REG+50              ;                                       6051
12061 042354  100031                            BPL     5$
12062 042356  104455                            TRAP    55                      ;                                       6054
12063 042360  000065                            .WORD   65
12064 042362  010464                            .WORD   ASYNC
12065 042364  024060                            .WORD   DUMPER
12066 042366  012746  006506                    MOV     #WRD.19,-(SP)           ;                                       6055
12067 042372  012746  007742                    MOV     #FNC.2,-(SP)
12068 042376  012746  006416                    MOV     #WRD.11,-(SP)
12069 042402  012746  007404                    MOV     #PHR.2,-(SP)
12070 042406  012746  006454                    MOV     #WRD.15,-(SP)
12071 042412  012746  006046                    MOV     #FOR.FMT,-(SP)
12072 042416  012746  000006                    MOV     #6,-(SP)
12073 042422  010600                            MOV     SP,R0                   ; SP,*
12074 042424  104414                            TRAP    14
12075 042426  104463                            TRAP    63
12076 042430  062706  000016            3$:     ADD     #16,SP                  ;                                       6051
12077 042434  000167  000546            4$:     JMP     12$                     ;                                       6053
12078 042440  017716  150600            5$:     MOV     @ML.REG+70,(SP)         ; *,ATA.SAVE                            6059
12079 042444  016701  151132                    MOV     ML.DUT,R1               ;                                       6061
12080 042450  006201                            ASR     R1
12081 042452  006201                            ASR     R1
```

```
12083                                ;ML4                                                    14-May-1981 13:48:54    TOPS
12084                                ;                                                       14-May-1981 13:05:08    PA:<
12085
12086 042454  006201                        ASR     R1
12087 042456  010600                        MOV     SP,R0                          ; ATA.SAVE,*
12088 042460  060001                        ADD     R0,R1
12089 042462  010146                        MOV     R1,-(SP)
12090 042464  016746  151112                MOV     ML.DUT,-(SP)
12091 042470  042716  177770                BIC     #177770,(SP)
12092 042474  012746  000001                MOV     #1,-(SP)
12093 042500  005046                        CLR     -(SP)
12094 042502  004767  140432                JSR     PC,BL$GT2
12095 042506  062706  000010                ADD     #10,SP
12096 042512  005300                        DEC     R0
12097 042514  001026                        BNE     6$
12098 042516  104455                        TRAP    55                             ;                6064
12099 042520  000072                        .WORD   72
12100 042522  010464                        .WORD   ASYNC
12101 042524  024060                        .WORD   DUMPER
12102 042526  012746  006506                MOV     #WRD.19,-(SP)                  ;                6065
12103 042532  012746  007742                MOV     #FNC.2,-(SP)
12104 042536  012746  006416                MOV     #WRD.11,-(SP)
12105 042542  012746  007404                MOV     #PHR.2,-(SP)
12106 042546  012746  006454                MOV     #WRD.15,-(SP)
12107 042552  012746  006062                MOV     #FIV.FMT,-(SP)
12108 042556  012746  000006                MOV     #6,-(SP)
12109 042562  010600                        MOV     SP,R0                          ; SP,*
12110 042564  104414                        TRAP    14
12111 042566  062706  000016                ADD     #16,SP                         ;                6063
12112 042572  104402                6$:     TRAP    2                              ;                6066
12113 042574  012777  000001  150432        MOV     #1,@ML.REG+60                  ;                6069
12114 042602  005077  150426                CLR     @ML.REG+60                     ;                6070
12115 042606  005016                        CLR     (SP)                           ; ATA.SAVE       6071
12116 042610  016701  150766                MOV     ML.DUT,R1                      ;                6072
12117 042614  006201                        ASR     R1
12118 042616  006201                        ASR     R1
12119 042620  006201                        ASR     R1
12120 042622  010600                        MOV     SP,R0                          ; ATA.SAVE,*
12121 042624  060001                        ADD     R0,R1
12122 042626  010146                        MOV     R1,-(SP)
12123 042630  016746  150746                MOV     ML.DUT,-(SP)
12124 042634  042716  177770                BIC     #177770,(SP)
12125 042640  012746  000001                MOV     #1,-(SP)
12126 042644  011646                        MOV     (SP),-(SP)
12127 042646  004767  140524                JSR     PC,BL$PU2
12128 042652  016677  000010  150364        MOV     10(SP),@ML.REG+70              ; ATA.SAVE,*     6073
12129 042660  005777  150340                TST     @ML.REG+50                     ;                6075
12130 042664  100026                        BPL     7$
12131 042666  104455                        TRAP    55                             ;                6078
12132 042670  000066                        .WORD   66
12133 042672  010464                        .WORD   ASYNC
12134 042674  024060                        .WORD   DUMPER
12135 042676  012746  010310                MOV     #REG.5,-(SP)                   ;                6079
12136 042702  012746  006470                MOV     #WRD.17,-(SP)
12137 042706  012746  006416                MOV     #WRD.11,-(SP)
```

```
12139                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
12140                                   ;                                                       14-May-1981 13:05:08    PA:<
12141
12142 042712  012746  007404                    MOV     #PHR.2,-(SP)
12143 042716  012746  006454                    MOV     #WRD.15,-(SP)
12144 042722  012746  006062                    MOV     #FIV.FMT,-(SP)
12145 042726  012746  000006                    MOV     #6,-(SP)
12146 042732  010600                            MOV     SP,R0                   ; SP,*
12147 042734  104414                            TRAP    14
12148 042736  062706  000016                    ADD     #16,SP                  ;                                       6077
12149 042742  062706  000010            7$:     ADD     #10,SP                  ;                                       6066
12150 042746  104467                            TRAP    67                      ;                                       6080
12151 042750  006000                            ROR     R0
12152 042752  103707                            BLO     6$
12153 042754  104402                    8$:     TRAP    2                       ;                                       6082
12154 042756  012777  000001  150250            MOV     #1,@ML.REG+60           ;                                       6084
12155 042764  005077  150244                    CLR     @ML.REG+60              ;                                       6085
12156 042770  012703  000001                    MOV     #1,R3                   ; *,DAT.PAT                             6086
12157 042774  005002                            CLR     R2                      ; ATA.SEL                               6088
12158 042776  020267  150600            9$:     CMP     R2,ML.DUT               ; ATA.SEL,*                             6091
12159 043002  001471                            BEQ     10$
12160 043004  010377  150234                    MOV     R3,@ML.REG+70           ; DAT.PAT,*                             6094
12161 043010  017716  150230                    MOV     @ML.REG+70,(SP)         ; *,ATA.SAVE                            6095
12162 043014  016701  150562                    MOV     ML.DUT,R1               ;                                       6097
12163 043020  006201                            ASR     R1
12164 043022  006201                            ASR     R1
12165 043024  006201                            ASR     R1
12166 043026  010600                            MOV     SP,R0                   ; ATA.SAVE,*
12167 043030  060001                            ADD     R0,R1
12168 043032  010146                            MOV     R1,-(SP)
12169 043034  016746  150542                    MOV     ML.DUT,-(SP)
12170 043040  042716  177770                    BIC     #177770,(SP)
12171 043044  012746  000001                    MOV     #1,-(SP)
12172 043050  005046                            CLR     -(SP)
12173 043052  004767  140062                    JSR     PC,BL$GT2
12174 043056  062706  000010                    ADD     #10,SP
12175 043062  005700                            TST     R0
12176 043064  001040                            BNE     10$
12177 043066  104455                            TRAP    55                      ;                                       6100
12178 043070  000067                            .WORD   67
12179 043072  010464                            .WORD   ASYNC
12180 043074  024060                            .WORD   DUMPER
12181 043076  012746  007516                    MOV     #PHR.7,-(SP)            ;                                       6101
12182 043102  012746  010310                    MOV     #REG.5,-(SP)
12183 043106  012746  006470                    MOV     #WRD.17,-(SP)
12184 043112  012746  006416                    MOV     #WRD.11,-(SP)
12185 043116  012746  007504                    MOV     #PHR.6,-(SP)
12186 043122  012746  006454                    MOV     #WRD.15,-(SP)
12187 043126  012746  006100                    MOV     #SIX.FMT,-(SP)
12188 043132  012746  000007                    MOV     #7,-(SP)
12189 043136  010600                            MOV     SP,R0                   ; SP,*
12190 043140  104414                            TRAP    14
12191 043142  010316                            MOV     R3,(SP)                 ; DAT.PAT,*                             6102
12192 043144  012746  004562                    MOV     #FMT.7,-(SP)
12193 043150  012746  000002                    MOV     #2,-(SP)
```

```
                                ;ML4                                                    14-May-1981 13:48:54    TOPS
                                ;                                                       14-May-1981 13:05:08    PA:<

043154  010600                          MOV     SP,R0                   ; SP,*
043156  104414                          TRAP    14
043160  062706  000024                  ADD     #24,SP                  ;                                       6103
043164  000405                          BR      11$
043166  006303                  10$:     ASL     R3                      ; DAT.PAT                               6108
043170  005202                          INC     R2                      ; ATA.SEL                               6088
043172  020227  000007                  CMP     R2,#7                   ; ATA.SEL,*
043176  003677                          BLE     9$
043200  104467                  11$:     TRAP    67                      ;                                       6109
043202  006000                          ROR     R0
043204  103663                          BLO     8$
043206  005726                  12$:     TST     (SP)+                   ;                                       5990
043210  000207                          RTS     PC

                                ; Routine Size:  302 words
                                ; Maximum stack depth per invocation:  16 words




                                        .SBTTL  T19
043212                          T19::
043212  004767  176640          1$:     JSR     PC,$T19                 ;                                       6111
043216  104466                          TRAP    66
043220  006000                          ROR     R0
043222  103773                          BLO     1$
043224  000207                          RTS     PC

                                ; Routine Size:  6 words
                                ; Maximum stack depth per invocation:  0 words


                        6113    !<BLF/PAGE>
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (70)

```
12245 ;ML4
12246 ;
12247
12248 ;          6114  !
12249 ;          6115  !
12250 ;          6116  BGNTST;
12251 ;          6117
12252 ;          6118  !++
12253 ;          6119  ! TEST NUMBER:  TST 20
12254 ;          6120  !
12255 ;          6121  ! TEST NAME:    SEARCH FUNCTION TEST
12256 ;          6122  !
12257 ;          6123  ! TEST DESCRIPTION:
12258 ;          6124  !
12259 ;          6125  !       TEST THE SEARCH FUNCTION BY:
12260 ;          6126  !
12261 ;          6127  !       1.  DOING A SEARCH FUNCTION AT
12262 ;          6128  !           ARRAY ZERO AND TEST GO,
12263 ;          6129  !           ERROR BITS AND ATTN FOR
12264 ;          6130  !           SETTING/NOT SETTING.
12265 ;          6131  !
12266 ;          6132  !       2.  DOING SEARCH FUNCTIONS AT ALL
12267 ;          6133  !           PRESENT ARRAYS' AND TEST ATTN
12268 ;          6134  !           SET
12269 ;          6135  !
12270 ;          6136  !       3.  DOING SEARCH FUNCTIONS AT ALL
12271 ;          6137  !           NOT PRESENT ARRAYS' AND TEST
12272 ;          6138  !           ATTN CLEARED.
12273 ;          6139  !
12274 ;          6140  !--
12275 ;          6141
12276 ;          6142  CLR_MBUS;
12277 ;          6143  MLDA = ZEROES;
12278 ;          6144  MLCS1 = SEARCH;                          !DO A SEARCH FUNCTION
12279 ;          6145
12280 ;          6146  if .GO IS_SET                            !SEE IF GO IS SET
12281 ;          6147  then
12282 ;          6148      begin
12283 ;          6149      ERRDF (56, ASYNC, DUMPER);           !ERROR IF NOT SET
12284 ;          6150      PRINTB (FIV_FMT, WRD_1, PHR_2, WRD_11, FNC_10, WRD_19);
12285 ;          6151      end;
12286 ;          6152
12287 ;          6153  if .ILF IS_SET                           !SEE IF ILF IS SET
12288 ;          6154  then
12289 ;          6155      begin
12290 ;          6156      ERRDF (57, ASYNC, DUMPER);           !ERROR IF SET
12291 ;          6157      PRINTB (FIV_FMT, WRD_3, PHR_5, WRD_11, FNC_11, FNC_10, WRD_19);
12292 ;          6158      end;
12293 ;          6159
12294 ;          6160  if .ATTN IS_NOT_SET                      !SEE IF ATTN IS SET
12295 ;          6161  then
12296 ;          6162      begin
12297 ;          6163      ERRDF (59, ASYNC, DUMPER);           !ERROR IF NOT SET
12298 ;          6164      PRINTB (FIV_FMT, WRD_16, PHR_1, WRD_12, FNC_10, WRD_19);
12299 ;          6165      end
```

```
12301 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
12302 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (70)
12303
12304 ;          6166  else
12305 ;          6167      begin
12306 ;          6168
12307 ;          6169      incr ARR_SEL from 0 to .LST_ARR by .ARR_INC do      !DO SEARCH AT ALL PRESENT ARRAYS
12308 ;          6170          begin
12309 ;          6171          BGNSUB;
12310 ;          6172          CLR_MBUS;
12311 ;          6173          MLDA = .ARR_SEL;                                !LOAD DSA REG WITH ARR_SEL
12312 ;          6174          MLCS1 = SEARCH;                                 !DO A SEARCH FUNCTION
12313 ;          6175
12314 ;          6176          if .OPI IS_SET                                  !READ ATTN
12315 ;          6177          then
12316 ;          6178              begin
12317 ;          6179              ERRDF (60, ASYNC, DUMPER);                  !ERROR IF NOT SET
12318 ;          6180              PRINTB (FIV_FMT, WRD_4, PHR_5, WRD_12, FNC_10, WRD_19);
12319 ;          6181              PRINTB (FMT_9, .ARR_SEL);
12320 ;          6182              end;
12321 ;          6183
12322 ;          6184          ENDSUB;
12323 ;          6185          end;
12324 ;          6186
12325 ;          6187      if .OP_NUM_ARR lss %o'000017'                       !SEE IF LSS 17 ARRAYS ARE PRESENT
12326 ;          6188      then
12327 ;          6189
12328 ;          6190          incr ARR_SEL from .LST_ARR + .ARR_INC to .ARR_16 by .ARR_INC do
12329 ;          6191                                                          !DO A SEARCH AT ALL NOT PRESENT
12330 ;          6192              begin                                       !ARRAYS IF LSS 17
12331 ;          6193              BGNSUB;
12332 ;          6194              CLR_MBUS;
12333 ;          6195              MLDA = .ARR_SEL;                            !LOAD DSA REG WITH ARR_SEL
12334 ;          6196              MLCS1 = SEARCH;                             !DO A SEARCH FUNCTION
12335 ;          6197
12336 ;          6198              if .OPI IS_NOT_SET                          !SEE IF OPI IS SET
12337 ;          6199              then
12338 ;          6200                  begin
12339 ;          6201                  ERRDF (61, ASYNC, DUMPER);              !ERROR IF NOT SET
12340 ;          6202                  PRINTB (FIV_FMT, WRD_4, PHR_1, WRD_12, FNC_10, WRD_19);
12341 ;          6203                  PRINTB (FMT_9, .ARR_SEL);
12342 ;          6204                  end;
12343 ;          6205
12344 ;          6206              ENDSUB;
12345 ;          6207              end;
12346 ;          6208
12347 ;          6209      end;
12348 ;          6210
12349 ;          6211  ENDTST;
12353
12354                                      .SBTTL  ST20
```

```
12362 043226  004167  140626          $T20:  JSR     R1,$SAVE3                    ;                6112
12363 043232  152777  000040  147754         BISB    #40,@ML.REG+40               ;                6116
12364 043240  016703  150336                 MOV     ML.DUT,R3
12365 043244  042703  177770                 BIC     #177770,R3
12366 043250  142777  000007  147736         BICB    #7,@ML.REG+40
12367 043256  150377  147732                 BISB    R3,@ML.REG+40
12368 043262  005077  147716                 CLR     @ML.REG+30                   ;                6143
12369 043266  012777  000031  147660         MOV     #31,@ML.REG                  ;                6144
12370 043274  132777  000001  147652         BITB    #1,@ML.REG                   ;                6146
12371 043302  001426                         BEQ     1$
12372 043304  104455                         TRAP    55                           ;                6149
12373 043306  000070                         .WORD   70
12374 043310  010464                         .WORD   ASYNC
12375 043312  024060                         .WORD   DUMPER
12376 043314  012746  006506                 MOV     #WRD.19,-(SP)                ;                6150
12377 043320  012746  010050                 MOV     #FNC.10,-(SP)
12378 043324  012746  006416                 MOV     #WRD.11,-(SP)
12379 043330  012746  007404                 MOV     #PHR.2,-(SP)
12380 043334  012746  006276                 MOV     #WRD.1,-(SP)
12381 043340  012746  006062                 MOV     #FIV.FMT,-(SP)
12382 043344  012746  000006                 MOV     #6,-(SP)
12383 043350  010600                         MOV     SP,R0                        ; SP,*
12384 043352  104414                         TRAP    14
12385 043354  062706  000016                 ADD     #16,SP                       ;                6148
12386 043360  132777  000001  147646  1$:    BITB    #1,@ML.REG+60                ;                6153
12387 043366  001430                         BEQ     2$
12388 043370  104455                         TRAP    55                           ;                6156
12389 043372  000071                         .WORD   71
12390 043374  010464                         .WORD   ASYNC
12391 043376  024060                         .WORD   DUMPER
12392 043400  012746  006506                 MOV     #WRD.19,-(SP)                ;                6157
12393 043404  012746  010050                 MOV     #FNC.10,-(SP)
12394 043410  012746  010060                 MOV     #FNC.11,-(SP)
12395 043414  012746  006416                 MOV     #WRD.11,-(SP)
12396 043420  012746  007472                 MOV     #PHR.5,-(SP)
12397 043424  012746  006314                 MOV     #WRD.3,-(SP)
12398 043430  012746  006062                 MOV     #FIV.FMT,-(SP)
12399 043434  012746  000007                 MOV     #7,-(SP)
12400 043440  010600                         MOV     SP,R0                        ; SP,*
12401 043442  104414                         TRAP    14
12402 043444  062706  000020                 ADD     #20,SP                       ;                6155
12403 043450  032777  100000  147546  2$:    BIT     #100000,@ML.REG+50           ;                6160
12404 043456  001027                         BNE     3$
12405 043460  104455                         TRAP    55                           ;                6163
12406 043462  000073                         .WORD   73
12407 043464  010464                         .WORD   ASYNC
12408 043466  024060                         .WORD   DUMPER
12409 043470  012746  006506                 MOV     #WRD.19,-(SP)                ;                6164
12410 043474  012746  010050                 MOV     #FNC.10,-(SP)
12411 043500  012746  006426                 MOV     #WRD.12,-(SP)
```

```
                                        ;ML4                                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                                       14-May-1981 13:05:08    PA:<
                                                
12416 043504  012746  007366                   MOV     #PHR.1,-(SP)
12417 043510  012746  006462                   MOV     #WRD.16,-(SP)
12418 043514  012746  006062                   MOV     #FIV.FMT,-(SP)
12419 043520  012746  000006                   MOV     #6,-(SP)
12420 043524  010600                          MOV     SP,R0                   ; SP,*
12421 043526  104414                          TRAP    14
12422 043530  062706  000016                   ADD     #16,SP                  ;                                       6162
12423 043534  000207                          RTS     PC                      ;                                       6160
12424 043536  016702  146034           3$:     MOV     LST.ARR,R2              ;                                       6169
12425 043542  016703  146014                   MOV     ARR.INC,R3
12426 043546  005001                          CLR     R1                      ; ARR.SEL
12427 043550  000467                          BR      6$
12428 043552  104402                   4$:    TRAP    2                       ;                                       6170
12429 043554  152777  000040  147432           BISB    #40,@ML.REG+40          ;                                       6171
12430 043562  016700  150014                   MOV     ML.DUT,R0
12431 043566  042700  177770                   BIC     #177770,R0
12432 043572  142777  000007  147414           BICB    #7,@ML.REG+40
12433 043600  150077  147410                   BISB    R0,@ML.REG+40
12434 043604  010177  147374                   MOV     R1,@ML.REG+30           ; ARR.SEL,*                             6173
12435 043610  012777  000031  147336           MOV     #31,@ML.REG             ;                                       6174
12436 043616  032777  020000  147410           BIT     #20000,@ML.REG+60       ;                                       6176
12437 043624  001435                          BEQ     5$
12438 043626  104455                          TRAP    55                      ;                                       6179
12439 043630  000074                          .WORD   74
12440 043632  010464                          .WORD   ASYNC
12441 043634  024060                          .WORD   DUMPER
12442 043636  012746  006506                   MOV     #WRD.19,-(SP)           ;                                       6180
12443 043642  012746  010050                   MOV     #FNC.10,-(SP)
12444 043646  012746  006426                   MOV     #WRD.12,-(SP)
12445 043652  012746  007472                   MOV     #PHR.5,-(SP)
12446 043656  012746  006322                   MOV     #WRD.4,-(SP)
12447 043662  012746  006062                   MOV     #FIV.FMT,-(SP)
12448 043666  012746  000006                   MOV     #6,-(SP)
12449 043672  010600                          MOV     SP,R0                   ; SP,*
12450 043674  104414                          TRAP    14
12451 043676  010116                          MOV     R1,(SP)                 ; ARR.SEL,*                             6181
12452 043700  012746  004644                   MOV     #FMT.9,-(SP)
12453 043704  012746  000002                   MOV     #2,-(SP)
12454 043710  010600                          MOV     SP,R0                   ; SP,*
12455 043712  104414                          TRAP    14
12456 043714  062706  000022                   ADD     #22,SP                  ;                                       6178
12457 043720  104467                   5$:    TRAP    67                      ;                                       6182
12458 043722  006000                          ROR     R0
12459 043724  103712                          BLO     4$
12460 043726  060301                          ADD     R3,R1                   ; *,ARR.SEL                             6169
12461 043730  020102                   6$:    CMP     R1,R2                   ; ARR.SEL,*
12462 043732  003707                          BLE     4$
12463 043734  026727  145620  000017           CMP     OP.NUM.ARR,#17          ;                                       6187
12464 043742  002102                          BGE     10$
12465 043744  016701  145626                   MOV     LST.ARR,R1              ;                                       6190
12466 043750  066701  145606                   ADD     ARR.INC,R1
12467 043754  016703  145614                   MOV     ARR.16,R3
```

```
12469                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
12470                                   ;                                                       14-May-1981 13:05:08    PA:<
12471
12472 043760  016702  145576                    MOV     ARR.INC,R2
12473 043764  000467                            BR      9$
12474 043766  104402                    7$:     TRAP    2                       ;                                       6192
12475 043770  152777  000040  147216            BISB    #40,@ML.REG+40          ;                                       6193
12476 043776  016700  147600                    MOV     ML.DUT,R0
12477 044002  042700  177770                    BIC     #177770,R0
12478 044006  142777  000007  147200            BICB    #7,@ML.REG+40
12479 044014  150077  147174                    BISB    R0,@ML.REG+40
12480 044020  010177  147160                    MOV     R1,@ML.REG+30           ; ARR.SEL,*                             6195
12481 044024  012777  000031  147122            MOV     #31,@ML.REG             ;                                       6196
12482 044032  032777  020000  147174            BIT     #20000,@ML.REG+60       ;                                       6198
12483 044040  001035                            BNE     8$
12484 044042  104455                            TRAP    55                      ;                                       6201
12485 044044  000075                            .WORD   75
12486 044046  010464                            .WORD   ASYNC
12487 044050  024060                            .WORD   DUMPER
12488 044052  012746  006506                    MOV     #WRD.19,-(SP)           ;                                       6202
12489 044056  012746  010050                    MOV     #FNC.10,-(SP)
12490 044062  012746  006426                    MOV     #WRD.12,-(SP)
12491 044066  012746  007366                    MOV     #PHR.1,-(SP)
12492 044072  012746  006322                    MOV     #WRD.4,-(SP)
12493 044076  012746  006062                    MOV     #FIV.FMT,-(SP)
12494 044102  012746  000006                    MOV     #6,-(SP)
12495 044106  010600                            MOV     SP,R0                   ; SP,*
12496 044110  104414                            TRAP    14
12497 044112  010116                            MOV     R1,(SP)                 ; ARR.SEL,*                             6203
12498 044114  012746  004644                    MOV     #FMT.9,-(SP)
12499 044120  012746  000002                    MOV     #2,-(SP)
12500 044124  010600                            MOV     SP,R0                   ; SP,*
12501 044126  104414                            TRAP    14
12502 044130  062706  000022                    ADD     #22,SP                  ;                                       6200
12503 044134  104467                    8$:     TRAP    67                      ;                                       6204
12504 044136  006000                            ROR     R0
12505 044140  103712                            BLO     7$
12506 044142  060201                            ADD     R2,R1                   ; *,ARR.SEL                             6190
12507 044144  020103                    9$:     CMP     R1,R3                   ; ARR.SEL,*
12508 044146  003707                            BLE     7$
12509 044150  000207                    10$:    RTS     PC                      ;                                       6112
12510
12511                                   ; Routine Size:  234 words
12512                                   ; Maximum stack depth per invocation:  13 words
12517
12518
12522
12523                                           .SBTTL  T20
```

```
12525                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
12526                                   ;                                                       14-May-1981 13:05:08    PA:<
12527
12531 044152                            T20::
12532 044152  004767  177050            1$:     JSR     PC,$T20                 ;                                       6209
12533 044156  104466                            TRAP    66
12534 044160  006000                            ROR     R0
12535 044162  103773                            BLO     1$
12536 044164  000207                            RTS     PC
12537
12538                                   ; Routine Size:  6 words
12539                                   ; Maximum stack depth per invocation:  0 words
12544
12545
12546 ;         6212  !<BLF/PAGE>
```

```
12548 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
12549 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (71)
12550
12551 ;          6213  !
12552 ;          6214  BGNTST;
12553 ;          6215
12554 ;          6216  !++
12555 ;          6217  ! TEST NUMBER:  TST 21
12556 ;          6218  !
12557 ;          6219  ! TEST NAME:    READ IN PRESET TEST
12558 ;          6220  !
12559 ;          6221  ! TEST DESCRIPTION:
12560 ;          6222  !       TEST THE READ IN PRESET FUNCTION BY:
12561 ;          6223  !
12562 ;          6224  !       1.  PERFORMING A READ-IN-PRESET FUNCTION AND TESTING GO, ERROR BITS
12563 ;          6225  !           AND VV FOR SET / NOT SET.
12564 ;          6226  !
12565 ;          6227  !--
12566 ;          6228
12567 ;          6229  CLR_MBUS;
12568 ;          6230  MLCS1 = RD_IN_PRE;                          !DO A READ IN PRESET FUNCTION
12569 ;          6231
12570 ;          6232  if .GO IS_SET                               !SEE IF GO IS NOT SET
12571 ;          6233  then
12572 ;          6234      begin
12573 ;          6235      ERRDF (62, ASYNC, DUMPER);              !ERROR IF SET
12574 ;          6236      PRINTB (FIV_FMT, WRD_1, PHR_2, WRD_11, FNC_11, WRD_19);
12575 ;          6237      end;
12576 ;          6238
12577 ;          6239  if .ILF IS_SET                              !SEE IF ILF IS NOT SET
12578 ;          6240  then
12579 ;          6241      begin
12580 ;          6242      ERRDF (63, ASYNC, DUMPER);              !ERROR IF SET
12581 ;          6243      PRINTB (FIV_FMT, WRD_3, PHR_5, WRD_11, FNC_11, WRD_19);
12582 ;          6244      end;
12583 ;          6245
12584 ;          6246  if .OPI IS_SET                              !SEE IF OPI IS NOT SET
12585 ;          6247  then
12586 ;          6248      begin
12587 ;          6249      ERRDF (64, ASYNC, DUMPER);              !ERROR IF SET
12588 ;          6250      PRINTB (FIV_FMT, WRD_4, PHR_5, WRD_11, FNC_11, WRD_19);
12589 ;          6251      end;
12590 ;          6252
12591 ;          6253  if .VV IS_NOT_SET                           !SEE IF VV IS SET
12592 ;          6254  then
12593 ;          6255      begin
12594 ;          6256      ERRDF (65, ASYNC, DUMPER);              !ERROR IF NOT SET
12595 ;          6257      PRINTB (FIV_FMT, WRD_18, PHR_1, WRD_11, FNC_11, WRD_19);
12596 ;          6258      end;
12597 ;          6259
12598 ;          6260  ENDTST;
12602
12603                              ;ML4                                   14-May-1981 13:48:54    TOPS
12604                              ;                                      14-May-1981 13:05:08    PA:<
12605
12606                                        .SBTTL  $T21
12610 044166  152777  000040  147020  $T21:  BISB    #40,@ML.REG+40          ;                      6214
```

```
12611 044174  016700  147402                MOV     ML.DUT,R0
12612 044200  042700  177770                BIC     #177770,R0
12613 044204  142777  000007  147002        BICB    #7,@ML.REG+40
12614 044212  150077  146776                BISB    R0,@ML.REG+40
12615 044216  012777  000021  146730        MOV     #21,@ML.REG             ;                       6230
12616 044224  132777  000001  146722        BITB    #1,@ML.REG              ;                       6232
12617 044232  001426                        BEQ     1$
12618 044234  104455                        TRAP    55                      ;                       6235
12619 044236  000076                        .WORD   76
12620 044240  010464                        .WORD   ASYNC
12621 044242  024060                        .WORD   DUMPER
12622 044244  012746  006506                MOV     #WRD.19,-(SP)           ;                       6236
12623 044250  012746  010060                MOV     #FNC.11,-(SP)
12624 044254  012746  006416                MOV     #WRD.11,-(SP)
12625 044260  012746  007404                MOV     #PHR.2,-(SP)
12626 044264  012746  006276                MOV     #WRD.1,-(SP)
12627 044270  012746  006062                MOV     #FIV.FMT,-(SP)
12628 044274  012746  000006                MOV     #6,-(SP)
12629 044300  010600                        MOV     SP,R0                   ; SP,*
12630 044302  104414                        TRAP    14
12631 044304  062706  000016                ADD     #16,SP                  ;                       6234
12632 044310  132777  000001  146716  1$:   BITB    #1,@ML.REG+60           ;                       6239
12633 044316  001426                        BEQ     2$
12634 044320  104455                        TRAP    55                      ;                       6242
12635 044322  000077                        .WORD   77
12636 044324  010464                        .WORD   ASYNC
12637 044326  024060                        .WORD   DUMPER
12638 044330  012746  006506                MOV     #WRD.19,-(SP)           ;                       6243
12639 044334  012746  010060                MOV     #FNC.11,-(SP)
12640 044340  012746  006416                MOV     #WRD.11,-(SP)
12641 044344  012746  007472                MOV     #PHR.5,-(SP)
12642 044350  012746  006314                MOV     #WRD.3,-(SP)
12643 044354  012746  006062                MOV     #FIV.FMT,-(SP)
12644 044360  012746  000006                MOV     #6,-(SP)
12645 044364  010600                        MOV     SP,R0                   ; SP,*
12646 044366  104414                        TRAP    14
12647 044370  062706  000016                ADD     #16,SP                  ;                       6241
12648 044374  032777  020000  146632  2$:   BIT     #20000,@ML.REG+60       ;                       6246
12649 044402  001426                        BEQ     3$
12650 044404  104455                        TRAP    55                      ;                       6249
12651 044406  000100                        .WORD   100
12652 044410  010464                        .WORD   ASYNC
12653 044412  024060                        .WORD   DUMPER
12654 044414  012746  006506                MOV     #WRD.19,-(SP)           ;                       6250
12655 044420  012746  010060                MOV     #FNC.11,-(SP)
12656 044424  012746  006416                MOV     #WRD.11,-(SP)
12657 044430  012746  007472                MOV     #PHR.5,-(SP)
12658                       ;ML4                                      14-May-1981 13:48:54    TOPS
12659                       ;                                         14-May-1981 13:05:08    PA:<
12660
12661 044434  012746  006322                MOV     #WRD.4,-(SP)
12662 044440  012746  006062                MOV     #FIV.FMT,-(SP)
12663 044444  012746  000006                MOV     #6,-(SP)
12664 044450  010600                        MOV     SP,R0                   ; SP,*
12665 044452  104414                        TRAP    14
12666 044454  062706  000016                ADD     #16,SP                  ;                       6248
12667 044460  132777  000100  146536  3$:   BITB    #100,@ML.REG+50         ;                       6253
```

```
12668 044466  001026                    BNE     4$
12669 044470  104455                    TRAP    55                              ;                       6256
12670 044472  000101                    .WORD   101
12671 044474  010464                    .WORD   ASYNC
12672 044476  024060                    .WORD   DUMPER
12673 044500  012746  006506            MOV     #WRD.19,-(SP)                   ;                       6257
12674 044504  012746  010060            MOV     #FNC.11,-(SP)
12675 044510  012746  006416            MOV     #WRD.11,-(SP)
12676 044514  012746  007366            MOV     #PHR.1,-(SP)
12677 044520  012746  006502            MOV     #WRD.18,-(SP)
12678 044524  012746  006062            MOV     #FIV.FMT,-(SP)
12679 044530  012746  000006            MOV     #6,-(SP)
12680 044534  010600                    MOV     SP,R0                           ; SP,*
12681 044536  104414                    TRAP    14
12682 044540  062706  000016            ADD     #16,SP                          ;                       6255
12683 044544  000207              4$:   RTS     PC                              ;                       6211
12684
12685                                   ; Routine Size:  120 words
12686                                   ; Maximum stack depth per invocation:  7 words
12691
12692
12696
12697                                           .SBTTL  T21
12701 044546                            T21::
12702 044546  004767  177414            1$:     JSR     PC,$T21                 ;                       6258
12703 044552  104466                            TRAP    66
12704 044554  006000                            ROR     R0
12705 044556  103773                            BLO     1$
12706 044560  000207                            RTS     PC
12707
12708                                   ; Routine Size:  6 words
12709                                   ; Maximum stack depth per invocation:  0 words
12717
12718
12719 ;          6261   !<BLF/PAGE>
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (72)

```
12721 ;ML4
12722 ;
12723
12724 ;          6262  !
12725 ;          6263  !
12726 ;          6264  BGNTST;
12727 ;          6265
12728 ;          6266  !++
12729 ;          6267  ! TEST NUMBER:  TST 22
12730 ;          6268  !
12731 ;          6269  ! TEST NAME:    ILLEGAL FUNCTION TEST
12732 ;          6270  !
12733 ;          6271  ! TEST DESCRIPTION:
12734 ;          6272  !
12735 ;          6273  !       TEST THE DETECTION OF ILLEGAL
12736 ;          6274  !       FUNCTIONS WRITTEN TO MLCS1
12737 ;          6275  !       BY:
12738 ;          6276  !
12739 ;          6277  !       WRITING ALL POSSIBLE ILLEGAL
12740 ;          6278  !       FUNCTIONS TO MLCS1.  THEN
12741 ;          6279  !       TEST GO AND ERROR BITS CLEARED.
12742 ;          6280  !
12743 ;          6281  !--
12744 ;          6282
12745 ;          6283  local
12746 ;          6284      BAD_BITS,                          !STORES A COUNT TO GENERATE BAD FUNCTIONS
12747 ;          6285      BAD_FUNC;                          !STORES GENERATED BAD FUNCTION
12748 ;          6286
12749 ;          6287  BAD_BITS = ZEROES;                     !CLEAR BAD BITS
12750 ;          6288
12751 ;          6289  incr CNT_1 from 0 to 2 do              !REPEAT LOOP 3 TIMES
12752 ;          6290      begin
12753 ;          6291      BAD_BITS = .BAD_BITS + %o'2';      !ADD 2 TO BAD_BITS
12754 ;          6292
12755 ;          6293      incr CNT_2 from %o'1' to %o'71' by %o'10' do      !REPEAT LOOP GENERATING 'GOOD' FUNCTIONS
12756 ;          6294          begin
12757 ;          6295          BGNSUB;
12758 ;          6296          CLR_MBUS;
12759 ;          6297          BAD_FUNC = .CNT_2 + .BAD_BITS;        !ADD BAD_BITS TO CNT_2 GENERATING BAD FUNCTIONS
12760 ;          6298          FIRST_BLK_XFER 7);                    !SET UP A FIRST BLOCK XFERR
12761 ;          6299          MLCS1 = .BAD_FUNC;                    !LOAD MLCS1 WITH TWO BAD FUNCTIONS
12762 ;          6300
12763 ;          6301          if .ILF IS_SET                        !SEE IF ILF IS SET
12764 ;          6302          then
12765 ;          6303              begin
12766 ;          6304
12767 ;          6305              if .GO IS_SET                     !IF ILF IS SET THEN SEE IF GO IS SET
12768 ;          6306              then
12769 ;          6307                  begin
12770 ;          6308                  ERRDF (66, ASYNC, DUMPER);    !ERROR IF GO SET WITH BAD FUNCTION
12771 ;          6309                  PRINTB (FIV_FMT, WRD_1, PHR_2, WRD_11, FNC_12, WRD_19);
12772 ;          6310                  PRINTB (FMT_12, .BAD_FUNC);
12773 ;          6311                  end;
12774 ;          6312
12775 ;          6313              end
```

```
12777 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
12778 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (72)
12779
12780 ;          6314          else
12781 ;          6315              begin
12782 ;          6316              ERRDF (67, ASYNC, DUMPER);          !ERROR BAD FUNCTION DON'T CAUSE ILF
12783 ;          6317              PRINTB (FIV_FMT, WRD_3, PHR_1, WRD_11, FNC_12, WRD_19);
12784 ;          6318              PRINTB (FMT_12, .BAD_FUNC);
12785 ;          6319              end;
12786 ;          6320
12787 ;          6321          if .OPI IS_SET                      !SEE IF OPI IS SET
12788 ;          6322          then
12789 ;          6323              begin
12790 ;          6324              ERRDF (68, ASYNC, DUMPER);          !ERROR IF SET
12791 ;          6325              PRINTB (FIV_FMT, WRD_4, PHR_5, WRD_12, FNC_12, WRD_19);
12792 ;          6326              PRINTB (FMT_12, .BAD_FUNC);
12793 ;          6327              end;
12794 ;          6328
12795 ;          6329          ENDSUB;
12796 ;          6330          end;
12797 ;          6331
12798 ;          6332      end;
12799 ;          6333
12800 ;          6334  ENDTST;
12804
12805                                  .SBTTL  $T22
12809 044562  004167  137330    $T22: JSR     R1,$SAVE5                 ;                                  6260
12810 044566  005004              CLR     R4                        ; BAD.BITS                         6287
12811 044570  005001              CLR     R1                        ; CNT.1                            6289
12812 044572  062704  000002    1$:   ADD     #2,R4                     ; *,BAD.BITS                       6291
12813 044576  012702  000001          MOV     #1,R2                     ; *,CNT.2                          6293
12814 044602  010405            2$:   MOV     R4,R5                     ; BAD.BITS,*                       6297
12815 044604  060205                  ADD     R2,R5                     ; CNT.2,*
12816 044606  104402            3$:   TRAP    2                         ;                                  6294
12817 044610  152777  000040  146376    BISB    #40,@ML.REG+40            ;                                  6295
12818 044616  016700  146760          MOV     ML.DUT,R0
12819 044622  042700  177770          BIC     #177770,R0
12820 044626  142777  000007  146360    BICB    #7,@ML.REG+40
12821 044634  150077  146354          BISB    R0,@ML.REG+40
12822 044640  010503                  MOV     R5,R3                     ; *,BAD.FUNC                       6297
12823 044642  004767  151172          JSR     PC,FIRST.BLK.XFER         ;                                  6298
12824 044646  010377  146302          MOV     R3,@ML.REG                ; BAD.FUNC,*                       6299
12825 044652  132777  000001  146354    BITB    #1,@ML.REG+60             ;                                  6301
12826 044660  001440                  BEQ     4$
12827 044662  132777  000001  146264    BITB    #1,@ML.REG                ;                                  6305
12828 044670  001471                  BEQ     6$
12829 044672  104455                  TRAP    55                        ;                                  6308
12830 044674  000102                  .WORD   102
12831 044676  010464                  .WORD   ASYNC
```

```
12833                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
12834                                   ;                                                       14-May-1981 13:05:08    PA:<
12835
12836 044700  024060                            .WORD   DUMPER
12837 044702  012746  006506                    MOV     #WRD.19,-(SP)           ;                                       6309
12838 044706  012746  010100                    MOV     #FNC.12,-(SP)
12839 044712  012746  006416                    MOV     #WRD.11,-(SP)
12840 044716  012746  007404                    MOV     #PHR.2,-(SP)
12841 044722  012746  006276                    MOV     #WRD.1,-(SP)
12842 044726  012746  006062                    MOV     #FIV.FMT,-(SP)
12843 044732  012746  000006                    MOV     #6,-(SP)
12844 044736  010600                            MOV     SP,R0                   ; SP,*
12845 044740  104414                            TRAP    14
12846 044742  010316                            MOV     R3,(SP)                 ; BAD.FUNC,*                            6310
12847 044744  012746  005004                    MOV     #FMT.12,-(SP)
12848 044750  012746  000002                    MOV     #2,-(SP)
12849 044754  010600                            MOV     SP,R0                   ; SP,*
12850 044756  104414                            TRAP    14
12851 044760  000433                            BR      5$                      ;                                       6307
12852 044762  104455                    4$:     TRAP    55                      ;                                       6316
12853 044764  000103                            .WORD   103
12854 044766  010464                            .WORD   ASYNC
12855 044770  024060                            .WORD   DUMPER
12856 044772  012746  006506                    MOV     #WRD.19,-(SP)           ;                                       6317
12857 044776  012746  010100                    MOV     #FNC.12,-(SP)
12858 045002  012746  006416                    MOV     #WRD.11,-(SP)
12859 045006  012746  007366                    MOV     #PHR.1,-(SP)
12860 045012  012746  006314                    MOV     #WRD.3,-(SP)
12861 045016  012746  006062                    MOV     #FIV.FMT,-(SP)
12862 045022  012746  000006                    MOV     #6,-(SP)
12863 045026  010600                            MOV     SP,R0                   , SP,*
12864 045030  104414                            TRAP    14
12865 045032  010316                            MOV     R3,(SP)                 ; BAD.FUNC,*                            6318
12866 045034  012746  005004                    MOV     #FMT.12,-(SP)
12867 045040  012746  000002                    MOV     #2,-(SP)
12868 045044  010600                            MOV     SP,R0                   ; SP,*
12869 045046  104414                            TRAP    14
12870 045050  062706  000022            5$:     ADD     #22,SP                  ;                                       6315
12871 045054  032777  020000  146152    6$:     BIT     #20000,@ML.REG+60       ;                                       6321
12872 045062  001435                            BEQ     7$
12873 045064  104455                            TRAP    55                      ;                                       6324
12874 045066  000104                            .WORD   104
12875 045070  010464                            .WORD   ASYNC
12876 045072  024060                            .WORD   DUMPER
12877 045074  012746  006506                    MOV     #WRD.19,-(SP)           ;                                       6325
12878 045100  012746  010100                    MOV     #FNC.12,-(SP)
12879 045104  012746  006426                    MOV     #WRD.12,-(SP)
12880 045110  012746  007472                    MOV     #PHR.5,-(SP)
12881 045114  012746  006322                    MOV     #WRD.4,-(SP)
12882 045120  012746  006062                    MOV     #FIV.FMT,-(SP)
12883 045124  012746  000006                    MOV     #6,-(SP)
12884 045130  010600                            MOV     SP,R0                   ; SP,*
12885 045132  104414                            TRAP    14
12886 045134  010316                            MOV     R3,(SP)                 ; BAD.FUNC,*                            6326
12887 045136  012746  005004                    MOV     #FMT.12,-(SP)
```

```
12889                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
12890                                   ;                                                       14-May-1981 13:05:08    PA:<
12891
12892 045142  012746  000002                    MOV     #2,-(SP)
12893 045146  010600                            MOV     SP,R0                   ; SP,*
12894 045150  104414                            TRAP    14
12895 045152  062706  000022                    ADD     #22,SP                  ;                                       6323
12896 045156  104467                    7$:     TRAP    67                      ;                                       6327
12897 045160  006000                            ROR     R0
12898 045162  103611                            BLO     3$
12899 045164  062702  000010                    ADD     #10,R2                  ; *,CNT.2                               6293
12900 045170  020227  000071                    CMP     R2,#71                  ; CNT.2,*
12901 045174  003602                            BLE     2$
12902 045176  005201                            INC     R1                      ; CNT.1                                 6289
12903 045200  020127  000002                    CMP     R1,#2                   ; CNT.1,*
12904 045204  003002                            BGT     8$
12905 045206  000167  177360                    JMP     1$
12906 045212  000207                    8$:     RTS     PC                      ;                                       6260
12907
12908                                   ; Routine Size:  141 words
12909                                   ; Maximum stack depth per invocation:  15 words
12914
12915
12919
12920                                           .SBTTL  T22
12924 045214                            T22::
12925 045214  004767  177342            1$:     JSR     PC,$T22                 ;                                       6332
12926 045220  104466                            TRAP    66
12927 045222  006000                            ROR     R0
12928 045224  103773                            BLO     1$
12929 045226  000207                            RTS     PC
12930
12931                                   ; Routine Size:  6 words
12932                                   ; Maximum stack depth per invocation:  0 words
12937
12938
12939 ;               6335  !<BLF/PAGE>
```

```
12941 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
12942 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (73)
12943
12944 ;          6336  !
12945 ;          6337  !
12946 ;          6338  BGNTST;
12947 ;          6339
12948 ;          6340  !++
12949 ;          6341  ! TEST NUMBER:  TST 23
12950 ;          6342  !
12951 ;          6343  ! TEST NAME:    REGISTER MODIFICATION REFUSED TEST
12952 ;          6344  !
12953 ;          6345  ! TEST DESCRIPTION:
12954 ;          6346  !
12955 ;          6347  !       TEST THE DETECTION OF A
12956 ;          6348  !       REGISTER MODIFICATION REFUSED
12957 ;          6349  !       BY:
12958 ;          6350  !
12959 ;          6351  !       1.  WRITTING TO MLCS1, MLDA
12960 ;          6352  !           AND MLER WHILE THE DRIVE
12961 ;          6353  !           IS BUSY AND TEST RMR
12962 ;          6354  !           BIT SET.
12963 ;          6355  !
12964 ;          6356  !       ALSO SEE IF THE DRIVE ASSERTED
12965 ;          6357  !       EXCEPTION BY TESTING THE TRE BIT SET.
12966 ;          6358  !
12967 ;          6359  !--
12968 ;          6360
12969 ;          6361  incr CNT from 0 to 2 do                  !REPEAT LOOP 3 TIMES
12970 ;          6362      begin
12971 ;          6363      BGNSUB;
12972 ;          6364      CLR_MBUS;
12973 ;          6365      MLCS1 = write;                       !DO A WRITE FUNCTION
12974 ;          6366
12975 ;          6367      case .CNT from 0 to 2 of             !WRITE TO SELECTED REGISTERS FORCING RMR
12976 ;          6368          set
12977 ;          6369
12978 ;          6370          [0] :
12979 ;          6371              MLCS1 = %o'000000';
12980 ;          6372
12981 ;          6373          [1] :
12982 ;          6374              MLDA = ONES;
12983 ;          6375
12984 ;          6376          [2] :
12985 ;          6377              MLER = ONES
12986 ;          6378          tes;
12987 ;          6379
12988 ;          6380      DELAY (FRTY_US);
12989 ;          6381
12990 ;          6382      if .RMR IS_NOT_SET                   !SEE IF RMR GOT SET
12991 ;          6383      then
12992 ;          6384          begin
12993 ;          6385          ERRDF (69, ASYNC, DUMPER);       !ERROR IF NOT SET
12994 ;          6386          PRINTB (FOR_FMT, WRD_21, PHR_1, WRD_11, WRD_21);
12995 ;          6387          end;
```

```
                                                                                14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
                                                                                14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (73)

                6388
                6389        if .TRE IS_NOT_SET                                  !SEE IF DRIVE ASSERTED EXCEPTION BY TESTING TRE
                6390        then
                6391            begin
                6392            ERRDF (117, SYNC, DUMPER);
                6393            PRINTB (FOR_FMT, WRD_20, PHR_1, WRD_11, WRD_21);
                6394            end;
                6395
                6396        ENDSUB;
                6397        end;
                6398
                6399    ENDTST;
```

```
                                        .SBTTL  $T23
045230  004167  136610          $T23:   JSR     R1,$SAVE2                       ;                               6334
045234  005746                          TST     -(SP)
045236  005002                          CLR     R2                              ; CNT                           6361
045240  104402                  1$:     TRAP    2                               ;                               6362
045242  152777  000040  145744          BISB    #40,@ML.REG+40                  ;                               6363
045250  016701  146326                  MOV     ML.DUT,R1
045254  042701  177770                  BIC     #177770,R1
045260  142777  000007  145726          BICB    #7,@ML.REG+40
045266  150177  145722                  BISB    R1,@ML.REG+40
045272  012777  000061  145654          MOV     #61,@ML.REG                     ;                               6365
045300  010201                          MOV     R2,R1                           ; CNT,*                         6367
045302  006301                          ASL     R1
045304  066107  045310                  ADD     2$(R1),PC
045310  000006                  2$:     .WORD   3$-2$
045312  000014                          .WORD   4$-2$
045314  000024                          .WORD   5$-2$
045316  005077  145632          3$:     CLR     @ML.REG                         ;                               6371
045322  000407                          BR      6$                              ;                               6367
045324  012777  177777  145652  4$:     MOV     #-1,@ML.REG+30                  ;                               6374
045332  000403                          BR      6$                              ;                               6367
045334  012777  177777  145672  5$:     MOV     #-1,@ML.REG+60                  ;                               6377
045342  012700  000050          6$:     MOV     #50,R0                          ; *,$STMP2                      6380
045346  001410                  7$:     BEQ     10$
045350  016701  134542                  MOV     L$DLY,R1                        ; *,$STMP1
045354  001403                          BEQ     9$
045356  005016                  8$:     CLR     (SP)                            ; $STMP
045360  005301                          DEC     R1                              ; $STMP1
045362  001375                          BNE     8$
045364  005300                  9$:     DEC     R0                              ; $STMP2
045366  000767                          BR      7$
045370  132777  000004  145636  10$:    BITB    #4,@ML.REG+60                   ;                               6382
045376  001024                          BNE     11$
```

```
13053                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
13054                                   ;                                                       14-May-1981 13:05:08    PA:<
13055
13056 045400  104455                            TRAP    55                      ;                                       6385
13057 045402  000105                            .WORD   105
13058 045404  010464                            .WORD   ASYNC
13059 045406  024060                            .WORD   DUMPER
13060 045410  012746  006522                    MOV     #WRD.21,-(SP)           ;                                       6386
13061 045414  012746  006416                    MOV     #WRD.11,-(SP)
13062 045420  012746  007366                    MOV     #PHR.1,-(SP)
13063 045424  012746  006522                    MOV     #WRD.21,-(SP)
13064 045430  012746  006046                    MOV     #FOR.FMT,-(SP)
13065 045434  012746  000005                    MOV     #5,-(SP)
13066 045440  010600                            MOV     SP,R0                   ; SP,*
13067 045442  104414                            TRAP    14
13068 045444  062706  000014                    ADD     #14,SP                  ;                                       6384
13069 045450  032777  040000  145476  11$:      BIT     #40000,@ML.REG          ;                                       6389
13070 045456  001024                            BNE     12$
13071 045460  104455                            TRAP    55                      ;                                       6392
13072 045462  000165                            .WORD   165
13073 045464  010526                            .WORD   SYNC
13074 045466  024060                            .WORD   DUMPER
13075 045470  012746  006522                    MOV     #WRD.21,-(SP)           ;                                       6393
13076 045474  012746  006416                    MOV     #WRD.11,-(SP)
13077 045500  012746  007366                    MOV     #PHR.1,-(SP)
13078 045504  012746  006514                    MOV     #WRD.20,-(SP)
13079 045510  012746  006046                    MOV     #FOR.FMT,-(SP)
13080 045514  012746  000005                    MOV     #5,-(SP)
13081 045520  010600                            MOV     SP,R0                   ; SP,*
13082 045522  104414                            TRAP    14
13083 045524  062706  000014                    ADD     #14,SP                  ;                                       6391
13084 045530  104467                    12$:    TRAP    67                      ;                                       6394
13085 045532  006000                            ROR     R0
13086 045534  103641                            BLO     1$
13087 045536  005202                            INC     R2                      ; CNT                                   6361
13088 045540  020227  000002                    CMP     R2,#2                   ; CNT,*
13089 045544  003635                            BLE     1$
13090 045546  005726                            TST     (SP)+                   ;                                       6334
13091 045550  000207                            RTS     PC
13092
13093                                   ; Routine Size:  105 words
13094                                   ; Maximum stack depth per invocation:  10 words
13099
13100
13104
13105                                           .SBTTL  T23
```

```
13113 045552                          T23::
13114 045552  004767  177452          1$:     JSR     PC,$T23                 ;      6397
13115 045556  104466                          TRAP    66
13116 045560  006000                          ROR     R0
13117 045562  103773                          BLO     1$
13118 045564  000207                          RTS     PC
13119
13120                                 ; Routine Size:  6 words
13121                                 ; Maximum stack depth per invocation:  0 words
13126
13127
13128 ;         6400  !<BLF/PAGE>
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (74)

```
13130 ;ML4
13131 ;
13132
13133 ;          6401  !
13134 ;          6402  !
13135 ;          6403  BGNTST;
13136 ;          6404
13137 ;          6405  !++
13138 ;          6406  ! TEST NUMBER:  TST 24
13139 ;          6407  !
13140 ;          6408  ! TEST NAME : initial PROM TEST
13141 ;          6409  !
13142 ;          6410  ! TEST DESCRIPTION:
13143 ;          6411  !
13144 ;          6412  !       TEST THE MEMORY ARRAYS' PROM
13145 ;          6413  !       TIMING AND CONTROL LOGIC FOR
13146 ;          6414  !       INITIAL PROM READS BY:
13147 ;          6415  !
13148 ;          6416  !       1.  READING 14 PROM LOCATIONS
13149 ;          6417  !           AND TESTING FOR:
13150 ;          6418  !
13151 ;          6419  !           A. CHECK SUM ERRORS AT
13152 ;          6420  !              EACH ROW COLUMN ADRS
13153 ;          6421  !
13154 ;          6422  !           B. SUM OF EACH PROM BIT SET
13155 ;          6423  !              <9,0> GTR 14.
13156 ;          6424  !
13157 ;          6425  !--
13158 ;          6426
13159 ;          6427  local
13160 ;          6428      OFF_SET_CNT : vector [10, byte],      !COUNTS EACH NIBBLE OFFSET
13161 ;          6429      ROW_ORED_COL : bitvector [16],        !SAVES ROW DATA OR'ED WITH COL DATA
13162 ;          6430      R_C_SAV : bitvector [16],             !TEMP LOCATION FOR ROW COL DATA
13163 ;          6431      PROM_ADRS,                            !PROM ADDRESS
13164 ;          6432      CHK_SUM,                              !CHECK SUM DATA
13165 ;          6433      CHK_SUM_ERR,                          !CHECK SUM ERROR
13166 ;          6434      BAD_NIB_CNT,                          !COUNTS BAD NIBBLES
13167 ;          6435      CNT_14_BAD,                           !COUNTS BAD NIBBLE POSITION EQL 14
13168 ;          6436      DODO_FLG,                             !DROP UNIT FLAG
13169 ;          6437      ERR_FLG;                              !ERROR FLAG
13170 ;          6438
13171 ;          6439  CLR_MBUS;
13172 ;          6440  DODO_FLG = ZERO;
13173 ;          6441  PROM_DIS = ONE;                           !SET PROM DISABLE MODE
13174 ;          6442  CHK_SUM_ERR = ZEROES;
13175 ;          6443
13176 ;          6444  incr CNT from 0 to 9 do                   !CLEAR OFFSET COUNTS
13177 ;          6445      OFF_SET_CNT [.CNT] = ZEROES;
13178 ;          6446
13179 ;          6447  incr ADRS_CNT from 0 to 14 do             !READ PROM DATA FROM 15 ARRAY WORDS
13180 ;          6448      begin
13181 ;          6449      ROW_ORED_COL = ZEROES;                !CLEAR ROW ORED COL SAVE LOCATION
13182 ;          6450      PROM_ADRS = .ADRS_CNT;                !GET COPY O' ADRS_CNT
13183 ;          6451
13184 ;          6452      incr TWICE from 0 to 1 do             !READ ROW AND COL DATA FOR THIS ARRAY WORD
```

```
                                                        14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
                                                        14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (74)

13186 ;ML4
13187 ;
13188 ;
13189 ;         6453        begin
13190 ;         6454        BAD_NIB_CNT = ZEROES;
13191 ;         6455        ERR_FLG = ZERO;
13192 ;         6456        MLPA = .PROM_ADRS;                  !LOADING MLPA INITIATES PROM READ
13193 ;         6457        DELAY (ONE_US);
13194 ;         6458        R_C_SAV = .MLPD;                    !GET THE ROW OR COL DATA
13195 ;         6459
13196 ;         6460        incr BIT_CNT from 0 to 9 do         !COUNT NUMBER BITS SET IN <9:0>
13197 ;         6461
13198 ;         6462            if .R_C_SAV [.BIT_CNT] IS_SET then BAD_NIB_CNT = .BAD_NIB_CNT + 1;
13199 ;         6463
13200 ;         6464        CHK_SUM = .R_C_SAV<10, 3>;          !GET THE CHECK SUM BITS
13201 ;         6465
13202 ;         6466        if .R_C_SAV [13] IS_SET then CHK_SUM = .CHK_SUM + 1;    !ADD IN BIT 13
13203 ;         6467
13204 ;         6468        if .R_C_SAV [14] IS_SET then CHK_SUM = .CHK_SUM + 1;    !ADD IN BIT 14
13205 ;         6469
13206 ;         6470        if .R_C_SAV [15] IS_SET then CHK_SUM = .CHK_SUM + 1;    !ADD IN BIT 15
13207 ;         6471
13208 ;         6472        if .R_C_SAV [15] IS_SET             !IS BIT 15 SET
13209 ;         6473        then
13210 ;         6474            begin
13211 ;         6475
13212 ;         6476            if .BAD_NIB_CNT lss .CHK_SUM then ERR_FLG = ONE;    !SET ERROR FLG IF CHECK SUM ERROR
13213 ;         6477
13214 ;         6478            end
13215 ;         6479        else
13216 ;         6480            begin
13217 ;         6481
13218 ;         6482            if .BAD_NIB_CNT neq .CHK_SUM then ERR_FLG = ONE;    !SET ERROR FLG IF CHECK SUM ERROR
13219 ;         6483
13220 ;         6484            end;
13221 ;         6485
13222 ;         6486        if .ERR_FLG IS_SET                  !WAS THERE A CHECK SUM ERROR
13223 ;         6487        then
13224 ;         6488            begin                           !REPORT INTERMEDIATE ERROR IF YES
13225 ;         6489            ERRDF (70, INTER, DUMPER);
13226 ;         6490            PRINTB (SIX_FMT, FNC_21, WRD_10, WRD_12, WRD_45, WRD_35, FNC_6);
13227 ;         6491            PRINTB (FMT_10, .CHK_SUM, .R_C_SAV);
13228 ;         6492            DODU_FLG = ONE;
13229 ;         6493            end;
13230 ;         6494
13231 ;         6495        ROW_ORED_COL = (.ROW_ORED_COL) or (.R_C_SAV);   !OR ROW AND COLUMN DATA
13232 ;         6496        PROM_ADRS = (.PROM_ADRS) or (%o'2000');!GET COLUMN DATA
13233 ;         6497        end;
13234 ;         6498
13235 ;         6499    incr index from 0 to 9 do               !GET TOTAL OFF SET COUNTS FROM THE 15 ARRAY WORDS
13236 ;         6500
13237 ;         6501        if .ROW_ORED_COL [.index] IS_SET then OFF_SET_CNT [.index] = .OFF_SET_CNT [.index] + 1;
13238 ;         6502
13239 ;         6503    end;
13240 ;         6504
```

```
13242 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
13243 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (74)
13244
13245 ;          6505  CNT_14_BAD = ZEROES;
13246 ;          6506
13247 ;          6507  incr CNT from 0 to 9 do                !COUNT HOW MANY OFFSET COUNTS GEQ 14
13248 ;          6508
13249 ;          6509      if .OFF_SET_CNT [.CNT] geq 14 then CNT_14_BAD = .CNT_14_BAD + 1;
13250 ;          6510
13251 ;          6511  if .CNT_14_BAD neq ZERO                !WAS ANY OFF_SET COUNTS GTR ZERO
13252 ;          6512  then
13253 ;          6513      begin
13254 ;          6514      ERRDF (71, ARR_DAT, DUMPER);       !ERROR IF YES
13255 ;          6515
13256 ;          6516      if .CNT_14_BAD eql 10              !WERE ALL 10 NIBBLE OFFSETS GEQ 14
13257 ;          6517      then
13258 ;          6518          PRINTB (SIX_FMT, WRD_55, WRD_54, WRD_46, WRD_51, WRD_42, WRD_40)        !ERROR IF YES
13259 ;          6519      else                               !IF NO THEN PRINT WHICH ONES WERE
13260 ;          6520          begin
13261 ;          6521          PRINTB (FOR_FMT, WRD_46, WRD_47, WRD_42, WRD_40);       !PRINT MESSAGE
13262 ;          6522
13263 ;          6523          incr CNT from 0 to 9 do            !FIND OFFSETS GEQ 14
13264 ;          6524
13265 ;          6525              if .OFF_SET_CNT [.CNT] geq 14 then PRINTB (FMT_13, .CNT, (.OFF_SET_CNT [.CNT]));
13266 ;          6526
13267 ;          6527                                             !PRINT NIBBLE POSITION AND COUNT
13268 ;          6528          end;
13269 ;          6529
13270 ;          6530      DODU_FLG = ONE;
13271 ;          6531      end;
13272 ;          6532
13273 ;          6533  if .DODU_FLG IS_SET                     !DROP THIS UNIT IF DODU_FLG SET
13274 ;          6534  then
13275 ;          6535      begin
13276 ;          6536      DODU (.ML_LUN);
13277 ;          6537      DOCLN;
13278 ;          6538      end;
13279 ;          6539
13280 ;          6540  ENDTST;
13284
13285                                          .SBTTL  $T24
13289 045566  004167  136324          $T24:   JSR     R1,$SAVE5                ;                    6399
13290 045572  162706  000030                  SUB     #30,SP
13291 045576  152777  000040  145410          BISB    #40,@ML.REG+40           ;                    6437
13292 045604  016705  145772                  MOV     ML.DUT,R5
13293 045610  042705  177770                  BIC     #177770,R5
13294 045614  142777  000007  145372          BICB    #7,@ML.REG+40
13295 045622  150577  145366                  BISB    R5,@ML.REG+40
13296 045626  005066  000002                  CLR     2(SP)                    ; DODU.FLG           6440
```

```
                                                                                                14-May-1981 13:48:54    TOPS
                                                                                                14-May-1981 13:05:08    PA:<
13298                                   ;ML4
13299                                   ;
13300
13301 045632  152777  000040  145434        BISB    #40,@ML.REG+120                                                 6441
13302 045640  005002                        CLR     R2              ; CHK.SUM.ERR                                   6442
13303 045642  012701  000016          1$:   MOV     #16,R1          ;                                               6445
13304 045646  060601                        ADD     SP,R1           ; OFF.SET.CNT,*
13305 045650  060201                        ADD     R2,R1           ; CNT,*
13306 045652  105011                        CLRB    (R1)
13307 045654  005202                        INC     R2              ; CNT                                           6444
13308 045656  020227  000011                CMP     R2,#11          ; CNT,*
13309 045662  003767                        BLE     1$
13310 045664  005005                        CLR     R5              ; ADRS.CNT                                      6447
13311 045666  005066  000010          2$:   CLR     10(SP)          ; ROW.ORED.COL                                  6449
13312 045672  010566  000006                MOV     R5,6(SP)        ; ADRS.CNT,PROM.ADRS                            6450
13313 045676  005004                        CLR     R4              ; TWICE                                         6452
13314 045700  005066  000004          3$:   CLR     4(SP)           ; BAD.NIB.CNT                                   6454
13315 045704  005016                        CLR     (SP)            ; ERR.FLG                                       6455
13316 045706  016677  000006  145340        MOV     6(SP),@ML.REG+100 ; PROM.ADRS,*                                 6456
13317 045714  012701  000001                MOV     #1,R1           ; *,$$TMP2                                      6457
13318 045720  001411                  4$:   BEQ     7$
13319 045722  016702  134170                MOV     L$DLY,R2        ; *,$$TMP1
13320 045726  001404                        BEQ     6$
13321 045730  005066  000014          5$:   CLR     14(SP)          ; $$TMP
13322 045734  005302                        DEC     R2              ; $$TMP1
13323 045736  001374                        BNE     5$
13324 045740  005301                  6$:   DEC     R1              ; $$TMP2
13325 045742  000766                        BR      4$
13326 045744  017766  145434  000012  7$:   MOV     @ML.REG+230,12(SP) ; *,R.C.SAV                                  6458
13327 045752  005002                        CLR     R2              ; BIT.CNT                                       6460
13328 045754  010201                  8$:   MOV     R2,R1           ; BIT.CNT,*                                     6462
13329 045756  006201                        ASR     R1
13330 045760  006201                        ASR     R1
13331 045762  006201                        ASR     R1
13332 045764  012700  000012                MOV     #12,R0
13333 045770  060600                        ADD     SP,R0           ; R.C.SAV,*
13334 045772  060001                        ADD     R0,R1
13335 045774  010146                        MOV     R1,-(SP)
13336 045776  010246                        MOV     R2,-(SP)        ; BIT.CNT,*
13337 046000  042716  177770                BIC     #177770,(SP)
13338 046004  012746  000001                MOV     #1,-(SP)
13339 046010  005046                        CLR     -(SP)
13340 046012  004767  135122                JSR     PC,BL$GT2
13341 046016  062706  000010                ADD     #10,SP
13342 046022  005300                        DEC     R0
13343 046024  001002                        BNE     9$
13344 046026  005266  000004                INC     4(SP)           ; BAD.NIB.CNT
13345 046032  005202                  9$:   INC     R2              ; BIT.CNT                                       6460
13346 046034  020227  000011                CMP     R2,#11          ; BIT.CNT,*
13347 046040  003745                        BLE     8$
13348 046042  016603  000012                MOV     12(SP),R3       ; R.C.SAV,CHK.SUM                               6464
13349 046046  006203                        ASR     R3              ; CHK.SUM
13350 046050  006203                        ASR     R3              ; CHK.SUM
13351 046052  000303                        SWAB    R3              ; CHK.SUM
13352 046054  042703  177770                BIC     #177770,R3      ; *,CHK.SUM
```

```
13354                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
13355                                   ;                                                       14-May-1981 13:05:08    PA:<
13356
13357 046060  132766  000040  000013          BITB    #40,13(SP)          ; *,R.C.SAV+1                                   6466
13358 046066  001401                          BEQ     10$
13359 046070  005203                          INC     R3                  ; CHK.SUM
13360 046072  132766  000100  000013  10$:    BITB    #100,13(SP)         ; *,R.C.SAV+1                                   6468
13361 046100  001401                          BEQ     11$
13362 046102  005203                          INC     R3                  ; CHK.SUM
13363 046104  005002                  11$:    CLR     R2                  ;                                               6470
13364 046106  105766  000013                  TSTB    13(SP)              ; R.C.SAV+1
13365 046112  100002                          BPL     12$
13366 046114  005202                          INC     R2
13367 046116  005203                          INC     R3                  ; CHK.SUM
13368 046120  006002                  12$:    ROR     R2                  ;                                               6472
13369 046122  103004                          BCC     13$
13370 046124  026603  000004                  CMP     4(SP),R3            ; BAD.NIB.CNT,CHK.SUM                           6476
13371 046130  002006                          BGE     15$
13372 046132  000403                          BR      14$
13373 046134  026603  000004          13$:    CMP     4(SP),R3            ; BAD.NIB.CNT,CHK.SUM                           6482
13374 046140  001402                          BEQ     15$
13375 046142  012716  000001          14$:    MOV     #1,(SP)             ; *,ERR.FLG
13376 046146  021627  000001          15$:    CMP     (SP),#1             ; ERR.FLG,*                                     6486
13377 046152  001044                          BNE     16$
13378 046154  104455                          TRAP    55                  ;                                               6489
13379 046156  000106                          .WORD   106
13380 046160  010672                          .WORD   INTER
13381 046162  024060                          .WORD   DUMPER
13382 046164  012746  010004                  MOV     #FNC.6,-(SP)        ;                                               6490
13383 046170  012746  006666                  MOV     #WRD.35,-(SP)
13384 046174  012746  006772                  MOV     #WRD.45,-(SP)
13385 046200  012746  006426                  MOV     #WRD.12,-(SP)
13386 046204  012746  006406                  MOV     #WRD.10,-(SP)
13387 046210  012746  010216                  MOV     #FNC.21,-(SP)
13388 046214  012746  006100                  MOV     #SIX.FMT,-(SP)
13389 046220  012746  000007                  MOV     #7,-(SP)
13390 046224  010600                          MOV     SP,R0               ; SP,*
13391 046226  104414                          TRAP    14
13392 046230  016616  000032                  MOV     32(SP),(SP)         ; R.C.SAV,*                                     6491
13393 046234  010346                          MOV     R3,-(SP)            ; CHK.SUM,*
13394 046236  012746  004676                  MOV     #FMT.10,-(SP)
13395 046242  012746  000003                  MOV     #3,-(SP)
13396 046246  010600                          MOV     SP,R0               ; SP,*
13397 046250  104414                          TRAP    14
13398 046252  012766  000001  000030          MOV     #1,30(SP)           ; *,DODU.FLG                                    6492
13399 046260  062706  000026                  ADD     #26,SP              ;                                               6488
13400 046264  056666  000012  000010  16$:    BIS     12(SP),10(SP)       ; R.C.SAV,ROW.ORED.COL                          6495
13401 046272  052766  002000  000006          BIS     #2000,6(SP)         ; *,PROM.ADRS                                   6496
13402 046300  005204                          INC     R4                  ; TWICE                                         6452
13403 046302  020427  000001                  CMP     R4,#1               ; TWICE,*
13404 046306  003002                          BGT     17$
13405 046310  000167  177364                  JMP     3$
13406 046314  005002                  17$:    CLR     R2                  ; INDEX                                         6499
13407 046316  010201                  18$:    MOV     R2,R1               ; INDEX,*                                       6501
13408 046320  006201                          ASR     R1
```

```
13410                          ;ML4                                                    14-May-1981 13:48:54    TOPS
13411                          ;                                                       14-May-1981 13:05:08    PA:<
13412
13413 046322  006201                    ASR     R1
13414 046324  006201                    ASR     R1
13415 046326  012700  000010            MOV     #10,R0
13416 046332  060600                    ADD     SP,R0                          ; ROW.ORED.COL,*
13417 046334  060001                    ADD     R0,R1
13418 046336  010146                    MOV     R1,-(SP)
13419 046340  010246                    MOV     R2,-(SP)                       ; INDEX,*
13420 046342  042716  177770            BIC     #177770,(SP)
13421 046346  012746  000001            MOV     #1,-(SP)
13422 046352  005046                    CLR     -(SP)
13423 046354  004767  134560            JSR     PC,BL$GT2
13424 046360  062706  000010            ADD     #10,SP
13425 046364  005300                    DEC     R0
13426 046366  001005                    BNE     19$
13427 046370  012701  000016            MOV     #16,R1
13428 046374  060601                    ADD     SP,R1                          ; OFF.SET.CNT,*
13429 046376  060201                    ADD     R2,R1                          ; INDEX,*
13430 046400  105211                    INCB    (R1)
13431 046402  005202          19$:      INC     R2                             ; INDEX                          6499
13432 046404  020227  000011            CMP     R2,#11                         ; INDEX,*
13433 046410  003742                    BLE     18$
13434 046412  005205                    INC     R5                             ; ADRS.CNT                       6447
13435 046414  020527  000016            CMP     R5,#16                         ; ADRS.CNT,*
13436 046420  003002                    BGT     20$
13437 046422  000167  177240            JMP     2$
13438 046426  005000          20$:      CLR     R0                             ; CNT.14.BAD                     6505
13439 046430  005001                    CLR     R1                             ; CNT                            6507
13440 046432  012702  000016  21$:      MOV     #16,R2                         ;                                6509
13441 046436  060602                    ADD     SP,R2                          ; OFF.SET.CNT,*
13442 046440  060102                    ADD     R1,R2                          ; CNT,*
13443 046442  121227  000016            CMPB    (R2),#16
13444 046446  103401                    BLO     22$
13445 046450  005200                    INC     R0                             ; CNT.14.BAD
13446 046452  005201          22$:      INC     R1                             ; CNT                            6507
13447 046454  020127  000011            CMP     R1,#11                         ; CNT,*
13448 046460  003764                    BLE     21$
13449 046462  005700                    TST     R0                             ; CNT.14.BAD                     6511
13450 046464  001505                    BEQ     27$
13451 046466  104455                    TRAP    55                             ;                                6514
13452 046470  000107                    .WORD   107
13453 046472  010570                    .WORD   ARR.DAT
13454 046474  024060                    .WORD   DUMPER
13455 046476  020027  000012            CMP     R0,#12                         ; CNT.14.BAD,*                   6516
13456 046502  001024                    BNE     23$
13457 046504  012746  006726            MOV     #WRD.40,-(SP)                  ;                                6518
13458 046510  012746  006746            MOV     #WRD.42,-(SP)
13459 046514  012746  007054            MOV     #WRD.51,-(SP)
13460 046520  012746  007004            MOV     #WRD.46,-(SP)
13461 046524  012746  007104            MOV     #WRD.54,-(SP)
13462 046530  012746  007114            MOV     #WRD.55,-(SP)
13463 046534  012746  006100            MOV     #SIX.FMT,-(SP)
13464 046540  012746  000007            MOV     #7,-(SP)
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                                MOV     SP,R0                   ; SP,*
                                                TRAP    14
                                                CMP     (SP)+,(SP)+
                                                BR      26$                     ;                                       6516
                                        23$:    MOV     #WRD.40,-(SP)           ;                                       6521
                                                MOV     #WRD.42,-(SP)
                                                MOV     #WRD.47,-(SP)
                                                MOV     #WRD.46,-(SP)
                                                MOV     #FOR.FMT,-(SP)
                                                MOV     #5,-(SP)
                                                MOV     SP,R0                   ; SP,*
                                                TRAP    14
                                                CLR     R2                      ; CNT                                   6523
                                        21$:    MOV     #32,R1                  ;                                       6525
                                                ADD     SP,R1                   ; OFF.SET.CNT,*
                                                ADD     R2,R1                   ; CNT,*
                                                CMPB    (R1),#16
                                                BLO     25$
                                                CLR     -(SP)
                                                MOVB    (R1),(SP)
                                                MOV     R2,-(SP)                ; CNT,*
                                                MOV     #FMT.13,-(SP)
                                                MOV     #3,-(SP)
                                                MOV     SP,R0                   ; SP,*
                                                TRAP    14
                                                ADD     #10,SP
                                        25$:    INC     R2                      ; CNT                                   6523
                                                CMP     R2,#11                  ; CNT,*
                                                BLE     24$
                                26$:            MOV     #1,16(SP)               ; *,DODU.FLG                            6530
                                                ADD     #14,SP                  ;                                       6513
                                27$:            CMP     2(SP),#1                ; DODU.FLG,*                            6533
                                                BNE     28$
                                                MOV     ML.LUN,R0               ;                                       6536
                                                TRAP    51
                                                TRAP    44
                                28$:            ADD     #30,SP                  ;                                       6399
                                                RTS     PC

                                        ; Routine Size:  304 words
                                        ; Maximum stack depth per invocation:  29 words


                                                .SBTTL  T24
```

```
13522                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
13523                                   ;                                                       14-May-1981 13:05:08    PA:<
13524
13528 046726                            T24::
13529 046726  004767  176634            1$:     JSR     PC,$T24                 ;                               6538
13530 046732  104466                            TRAP    66
13531 046734  006000                            ROR     R0
13532 046736  103773                            BLO     1$
13533 046740  000207                            RTS     PC
13534
13535                                   ; Routine Size:  6 words
13536                                   ; Maximum stack depth per invocation:  0 words
13541
13542
13543 ;         6541  !<BLF/PAGE>
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (75)

```
13545 ;ML4
13546 ;
13547
13548 ;          6542  !
13549 ;          6543  !
13550 ;          6544  BGNTST;
13551 ;          6545
13552 ;          6546  !++
13553 ;          6547  ! TEST NUMBER:  TST 25
13554 ;          6548  !
13555 ;          6549  ! TEST NAME:    PROM OR FUNCTION TEST
13556 ;          6550  !
13557 ;          6551  ! TEST DESCRIPTION:
13558 ;          6552  !
13559 ;          6553  !       TEST THE HARDWARE ORING OF
13560 ;          6554  !       THE MEMORY ARRAYS' PROM
13561 ;          6555  !       ROW COLUMN DATA BY:
13562 ;          6556  !
13563 ;          6557  !       1.  READING AND STORING 128
13564 ;          6558  !           HARDWARE ORED ROW COLUMN
13565 ;          6559  !           DATA.
13566 ;          6560  !
13567 ;          6561  !       2.  THEN IN PROM DISABLE MODE
13568 ;          6562  !           AND VIA SOFTWARE CONTROL,
13569 ;          6563  !           READ AND OR PROM ROW
13570 ;          6564  !           COLUMN DATA AND COMPARE
13571 ;          6565  !           AGAINST THE RESPECTIVE
13572 ;          6566  !           STORED HARDWARE ORED DATA.
13573 ;          6567  !
13574 ;          6568  !--
13575 ;          6569
13576 ;          6570  local
13577 ;          6571      R_BITS,                          !PROM ROW DATA
13578 ;          6572      C_BITS,                          !PROM COL DATA
13579 ;          6573      SW_ORED,                         !SOFTWARE CALCULATED PROM ORED DATA
13580 ;          6574      HW_SAVE,                         !SOFTWARE  PROM ORED DATA
13581 ;          6575      DODU_FLG;                        !DROP UNIT FLAG
13582 ;          6576
13583 ;          6577  BGNSUB;
13584 ;          6578  CLR_MBUS;
13585 ;          6579  DODU_FLG = ZERO;
13586 ;          6580  DAT_DM = ONE;
13587 ;          6581  FIRST_BLK_XFER ();                   !SET UP A FIRST BLOCK XFER
13588 ;          6582  ML_FUNC = write;                     !DO A WRITE FUNCTION
13589 ;          6583
13590 ;          6584  incr PROM_ADRS from 0 to 127 do      !READ AND STORE 128 HARDWARE PROM ORED DATA
13591 ;          6585      begin
13592 ;          6586      DELAY (ONE_US);
13593 ;          6587      HW_OR_TBL [.PROM_ADRS] = .MLPD;  !READ HARDWARE PROM ORED DATA
13594 ;          6588      DAT_CLK = ONE;                   !CLOCK NEXT ONE OUT
13595 ;          6589      end;
13596 ;          6590
13597 ;          6591  CLR_MBUS;
13598 ;          6592  PROM_DIS = ONE;                      !SET PROM DISABLE MODE
13599 ;          6593
```

```
 13601 ;ML4                                                          14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
 13602 ;                                                             14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (75)
 13603
 13604 ;          6594  incr PROM_ADRS from 0 to 127 do             !CALCULATE 128 SW ORED DATA & COMPARE TO HW TABLE
 13605 ;          6595      begin
 13606 ;          6596      MLPA = .PROM_ADRS;                      !LOADING MLPA INITIATES A PROM READ
 13607 ;          6597      DELAY (ONE_US);
 13608 ;          6598      R_BITS = .MLPD;                         !SAVE ROW DATA
 13609 ;          6599      MLPA = .PROM_ADRS or %o'2000';          !ENABLE COLUMN DATA ADRS
 13610 ;          6600      DELAY (ONE_US);
 13611 ;          6601      C_BITS = .MLPD;                         !SAVE COL DATA
 13612 ;          6602      SW_ORED = .R_BITS or .C_BITS;           !CALCULATE SOFTWARE ORED
 13613 ;          6603      HW_SAVE = .HW_OR_TBL [.PROM_ADRS];      !GET RESPECTIVE HARDWARE ORED
 13614 ;          6604
 13615 ;          6605      if (.HW_SAVE<0, 9>) neq (.SW_ORED<0, 9>)    !COMPARE SW & HW ORED
 13616 ;          6606      then
 13617 ;          6607          begin                               !IF NEQ THEN ERROR
 13618 ;          6608          ERRDF (76, ARR_DAT, DUMPER);
 13619 ;          6609          PRINTB (FOR_FMT, WRD_35, WRD_36, WRD_19, PHR_4);
 13620 ;          6610          PRINTB (FMT_2, .SW_ORED<0, 9>, .HW_SAVE<0, 9>, (.SW_ORED<0, 9> xor .HW_SAVE<0, 9>));
 13621 ;          6611          DODU_FLG = ONE;
 13622 ;          6612          end;
 13623 ;          6613
 13624 ;          6614      end;
 13625 ;          6615
 13626 ;          6616  ENDSUB;
 13627 ;          6617  if .DODU_FLG IS_SET                          !DROP THIS UNIT IF DODU FLG IS_SET
 13628 ;          6618  then
 13629 ;          6619      begin
 13630 ;          6620      DODU (.ML_LUN);
 13631 ;          6621      DOCLN;
 13632 ;          6622      end;
 13633 ;          6623
 13634 ;          6624  ENDTST;
 13638
 13639                                          .SBTTL  $T25
 13643 046742  004167  135150          $T25:   JSR     R1,$SAVE5                ;                          6540
 13644 046746  162706  000010                  SUB     #10,SP
 13645 046752  104402                  1$:     TRAP    2                        ;                          6575
 13646 046754  152777  000040  144232          BISB    #40,@ML.REG+40           ;                          6577
 13647 046762  016705  144614                  MOV     ML.DUT,R5
 13648 046766  042705  177770                  BIC     #177770,R5
 13649 046772  142777  000007  144214          BICB    #7,@ML.REG+40
 13650 047000  150577  144210                  BISB    R5,@ML.REG+40
 13651 047004  005016                          CLR     (SP)                     ; DODU.FLG                 6579
 13652 047006  152777  000010  144260          BISB    #10,@ML.REG+120          ;                          6580
 13653 047014  004767  147020                  JSR     PC,FIRST.BLK.XFER        ;                          6581
 13654 047020  142777  000077  144126          BICB    #77,@ML.REG              ;                          6582
 13655 047026  152777  000061  144120          BISB    #61,@ML.REG
```

```
                                                                                14-May-1981 13:48:54    TOPS
                                                                                14-May-1981 13:05:08    PA:<
13657                           ;ML4
13658                           ;
13659
13660 047034  005004                    CLR     R4                      ; PROM.ADRS                     6584
13661 047036  012703  000001      2$:     MOV     #1,R3                   ; *,$$TMP2                      6586
13662 047042  001411              3$:     BEQ     6$
13663 047044  016705  133046              MOV     L$DLY,R5                ; *,$$TMP1
13664 047050  001404                    BEQ     5$
13665 047052  005066  000006      4$:     CLR     6(SP)                   ; $$TMP
13666 047056  005305                    DEC     R5                      ; $$TMP1
13667 047060  001374                    BNE     4$
13668 047062  005303              5$:     DEC     R3                      ; $$TMP2
13669 047064  000766                    BR      3$
13670 047066  010405              6$:     MOV     R4,R5                   ; PROM.ADRS,*                   6587
13671 047070  006305                    ASL     R5
13672 047072  017765  144306  011160      MOV     @ML.REG+230,HW.OR.TBL(R5)
13673 047100  152777  000020  144166      BISB    #20,@ML.REG+120         ;                               6588
13674 047106  005204                    INC     R4                      ; PROM.ADRS                     6584
13675 047110  020427  000177            CMP     R4,#177                 ; PROM.ADRS,*
13676 047114  003750                    BLE     2$
13677 047116  152777  000040  144070      BISB    #40,@ML.REG+40          ;                               6589
13678 047124  016705  144452              MOV     ML.DUT,R5
13679 047130  042705  177770              BIC     #177770,R5
13680 047134  142777  000007  144052      BICB    #7,@ML.REG+40
13681 047142  150577  144046              BISB    R5,@ML.REG+40
13682 047146  152777  000040  144120      BISB    #40,@ML.REG+120         ;                               6592
13683 047154  005003                    CLR     R3                      ; PROM.ADRS                     6594
13684 047156  010377  144072      7$:     MOV     R3,@ML.REG+100          ; PROM.ADRS,*                   6596
13685 047162  012704  000001              MOV     #1,R4                   ; *,$$TMP2                      6597
13686 047166  001411              8$:     BEQ     11$
13687 047170  016705  132722              MOV     L$DLY,R5                ; *,$$TMP1
13688 047174  001404                    BEQ     10$
13689 047176  005066  000006      9$:     CLR     6(SP)                   ; $$TMP
13690 047202  005305                    DEC     R5                      ; $$TMP1
13691 047204  001374                    BNE     9$
13692 047206  005304              10$:    DEC     R4                      ; $$TMP2
13693 047210  000766                    BR      8$
13694 047212  017766  144166  000004  11$:     MOV     @ML.REG+230,4(SP)       ; *,R.BITS                      6598
13695 047220  010305                    MOV     R3,R5                   ; PROM.ADRS,*                   6599
13696 047222  052705  002000              BIS     #2000,R5
13697 047226  010577  144022              MOV     R5,@ML.REG+100
13698 047232  012704  000001              MOV     #1,R4                   ; *,$$TMP2                      6600
13699 047236  001411              12$:    BEQ     15$
13700 047240  016705  132652              MOV     L$DLY,R5                ; *,$$TMP1
13701 047244  001404                    BEQ     14$
13702 047246  005066  000006      13$:    CLR     6(SP)                   ; $$TMP
13703 047252  005305                    DEC     R5                      ; $$TMP1
13704 047254  001374                    BNE     13$
13705 047256  005304              14$:    DEC     R4                      ; $$TMP2
13706 047260  000766                    BR      12$
13707 047262  017766  144116  000002  15$:     MOV     @ML.REG+230,2(SP)       ; *,C.BITS                      6601
13708 047270  016601  000004              MOV     4(SP),R1                ; R.BITS,SW.ORED                6602
13709 047274  056601  000002              BIS     2(SP),R1                ; C.BITS,SW.ORED
13710 047300  010305                    MOV     R3,R5                   ; PROM.ADRS,*                   6603
13711 047302  006305                    ASL     R5
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                        ;
013716 047304  016502  011160                 MOV     HW.OR.TBL(R5),R2        ; *,HW.SAVE
013717 047310  010104                         MOV     R1,R4                   ; SW.ORED,*                                     6605
013718 047312  042704  177000                 BIC     #177000,R4
013719 047316  010205                         MOV     R2,R5                   ; HW.SAVE,*
013720 047320  042705  177000                 BIC     #177000,R5
013721 047324  020504                         CMP     R5,R4
013722 047326  001451                         BEQ     16$
013723 047330  104455                         TRAP    55                      ;                                               6608
013724 047332  000114                         .WORD   114
013725 047334  010570                         .WORD   ARR.DAT
013726 047336  024060                         .WORD   DUMPER
013727 047340  012746  007454                 MOV     #PHR.4,-(SP)            ;                                               6609
013728 047344  012746  006506                 MOV     #WRD.19,-(SP)
013729 047350  012746  006674                 MOV     #WRD.36,-(SP)
013730 047354  012746  006666                 MOV     #WRD.35,-(SP)
013731 047360  012746  006046                 MOV     #FOR.FMT,-(SP)
013732 047364  012746  000005                 MOV     #5,-(SP)
013733 047370  010600                         MOV     SP,R0                   ; SP,*
013734 047372  104414                         TRAP    14
013735 047374  010205                         MOV     R2,R5                   ; HW.SAVE,*                                     6610
013736 047376  010116                         MOV     R1,(SP)                 ; SW.ORED,*
013737 047400  010504                         MOV     R5,R4
013738 047402  040104                         BIC     R1,R4
013739 047404  040516                         BIC     R5,(SP)
013740 047406  050416                         BIS     R4,(SP)
013741 047410  010246                         MOV     R2,-(SP)                ; HW.SAVE,*
013742 047412  042716  177000                 BIC     #177000,(SP)
013743 047416  010146                         MOV     R1,-(SP)                ; SW.ORED,*
013744 047420  042716  177000                 BIC     #177000,(SP)
013745 047424  012746  004266                 MOV     #FMT.2,-(SP)
013746 047430  012746  000004                 MOV     #4,-(SP)
013747 047434  010600                         MOV     SP,R0                   ; SP,*
013748 047436  104414                         TRAP    14
013749 047440  012766  000001  000024         MOV     #1,24(SP)               ; *,DODU.FLG                                    6611
013750 047446  062706  000024                 ADD     #24,SP                  ;                                               6607
013751 047452  005203                 16$:    INC     R3                      ; PROM.ADRS                                     6594
013752 047454  020327  000177                 CMP     R3,#177                 ; PROM.ADRS,*
013753 047460  003636                         BLE     7$
013754 047462  104467                         TRAP    67                      ;                                               6614
013755 047464  006000                         ROR     R0
013756 047466  103002                         BHIS    17$
013757 047470  000167  177256                 JMP     1$
013758 047474  021627  000001         17$:    CMP     (SP),#1                 ; DODU.FLG,*                                    6617
013759 047500  001004                         BNE     18$
013760 047502  016700  144072                 MOV     ML.LUN,R0               ;                                               6620
013761 047506  104451                         TRAP    51
013762 047510  104444                         TRAP    44
013763 047512  062706  000010         18$:    ADD     #10,SP                  ;                                               6540
013764 047516  000207                         RTS     PC
013765
013766                                  ; Routine Size:  183 words
013767                                  ; Maximum stack depth per invocation:  20 words
```

```
                                ;ML4                                                    14-May-1981 13:48:54    TOPS
                                ;                                                       14-May-1981 13:05:08    PA:<
                                .SBTTL  T25
                        T25::
                        1$:     JSR     PC,$T25                 ;                                               6622
                                TRAP    66
                                ROR     R0
                                BLO     1$
                                RTS     PC

                                ; Routine Size:  6 words
                                ; Maximum stack depth per invocation:  0 words

;               6625    !<BLF/PAGE>
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (76)

```
13803 ;ML4
13804 ;
13805
13806 ;          6626  !
13807 ;          6627  !
13808 ;          6628  BGNTST;
13809 ;          6629
13810 ;          6630  !++
13811 ;          6631  ! TEST NUMBER:  TST 26
13812 ;          6632  !
13813 ;          6633  ! TEST NAME:    UV ADRS ERROR TEST
13814 ;          6634  !
13815 ;          6635  ! TEST DESCRIPTION:
13816 ;          6636  !
13817 ;          6637  !       TEST THE DETECTION OF UV ADRS
13818 ;          6638  !       ERRORS BY:
13819 ;          6639  !
13820 ;          6640  !       1.  GENERATING PROM DATA PATTERN
13821 ;          6641  !           FROM 0 TO %O'177777' AND
13822 ;          6642  !           DETERMINE WHETHER RESPECTIVE
13823 ;          6643  !           PATTERN IS GOOD/OR BAD
13824 ;          6644  !           PROM DATA.
13825 ;          6645  !
13826 ;          6646  !       2.  VIA DAT_DM AND PROM R/W
13827 ;          6647  !           MODES PRESENT GENERATED
13828 ;          6648  !           PROM DATA TO THE UV ADRS
13829 ;          6649  !           ERR PROM.
13830 ;          6650  !
13831 ;          6651  !       3.  TEST ERROR CONDITIONS FOR
13832 ;          6652  !           CORRECT RESPONCE TO GOOD/
13833 ;          6653  !           OR BAD PROM DATA.
13834 ;          6654  !
13835 ;          6655  !--
13836 ;          6656
13837 ;          6657  local
13838 ;          6658      DODU_FLG,                       !DROP UNIT FLAG
13839 ;          6659      HIGH_CNT,                       !STORES PROM DATA CHECK SUM BITS
13840 ;          6660      PROM_DATA : bitvector [16],     !STORES PROM DATA
13841 ;          6661      LOW_CNT,                        !STORES SUM OF PROM DATA BITS <9:0>
13842 ;          6662      TEMP,                           !TEMPORARY STORAGE
13843 ;          6663      ERR_FLG,                        !ERROR FLAG
13844 ;          6664      GTR_FLG;                        !SETS WHEN PROM DATA BIT 15 IS A ONE
13845 ;          6665
13846 ;          6666  DODU_FLG = ZERO;
13847 ;          6667  PROM_DATA = -1;
13848 ;          6668
13849 ;          6669  do                                  !TEST ALL POSSIBLE PROM DATA COMBINATIONS
13850 ;          6670      begin
13851 ;          6671      PROM_DATA = .PROM_DATA + 1;     !INCREMENT PROM_DATA
13852 ;          6672      BGNSDB;
13853 ;          6673      CLR_MBUS;
13854 ;          6674      ERR_FLG = ZERO;
13855 ;          6675      LOW_CNT = ZEROES;
13856 ;          6676      HIGH_CNT = ZEROES;
13857 ;          6677      GTR_FLG = ZERO;
```

```
13859 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
13860 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (76)
13861
13862 ;          6678
13863 ;          6679        incr BIT_CNT from 0 to 9 do                  !COUNT PROM DATA BITS <9:0>
13864 ;          6680
13865 ;          6681            if .PROM_DATA [.BIT_CNT] IS_SET then LOW_CNT = .LOW_CNT + 1;
13866 ;          6682
13867 ;          6683        HIGH_CNT = .PROM_DATA<10, 3>;                !GET PROM DATA CHECK SUM BITS
13868 ;          6684
13869 ;          6685        if .PROM_DATA [13] IS_SET then HIGH_CNT = .HIGH_CNT + 1;    !ADD IN BIT 13 IF SET
13870 ;          6686
13871 ;          6687        if .PROM_DATA [14] IS_SET then HIGH_CNT = .HIGH_CNT + 1;    !ADD IN BIT 14 IF SET
13872 ;          6688
13873 ;          6689        if .PROM_DATA [15] IS_SET then HIGH_CNT = .HIGH_CNT + 1;    !ADD IN BIT 15 IF SET
13874 ;          6690
13875 ;          6691        if .PROM_DATA [15] IS_SET then GTR_FLG = ONE;       !SET FLAG IF BIT 15 IS SET
13876 ;          6692
13877 ;          6693        DAT_DM_XFER ();                              !SET UP A DATA DIAG MODE XFERR
13878 ;          6694        PROM_RW = ONE;                               !SET PROM READ WRITE
13879 ;          6695        MLPD = .PROM_DATA;                           !LOAD MLPD WITH PROM_DATA
13880 ;          6696        MLCS1 = write;                               !DO A WRITE FUNCTION
13881 ;          6697        DAT_CLK = ONE;                               !CLOCK PROM DATA INTO UV PROM
13882 ;          6698
13883 ;          6699        if .UNS IS_SET                               !SEE IF PROM DATA CAUSED A UV ERROR
13884 ;          6700        then
13885 ;          6701            begin
13886 ;          6702
13887 ;          6703            if .GTR_FLG IS_SET                       !UNS IS_SET. SEE IF GTR FLG IS SET
13888 ;          6704            then
13889 ;          6705                begin
13890 ;          6706                TEMP = .PROM_DATA;                   !LOAD TEMP WITH PROM DATA
13891 ;          6707                TEMP = .TEMP and %o'162000';         !SEE IF THESE BITS ARE SET IN PROM DATA
13892 ;          6708
13893 ;          6709                if .TEMP eql %o'162000'              !THESE BITS SET AUTOMATICALLY CAUSE A UNS
13894 ;          6710                then
13895 ;          6711                    begin
13896 ;          6712
13897 ;          6713                    if .LOW_CNT geq .HIGH_CNT        !LOW<9:0> SHOULD BE ISS THAN THE HIGH<15:10> IF GTR FLG IS S
13898 ;          6714                    then
13899 ;          6715                        begin                        !ERROR IF LOW<9:0> IS GEQ HIGH<15:10>
13900 ;          6716                        ERRDF (72, ARR_DAT, DUMPER);
13901 ;          6717                        PRINTB (SIX_FMT, WRD_34, PHR_5, WRD_32, WRD_6, WRD_33, WRD_24);
13902 ;          6718                        ERR_FLG = ONE;
13903 ;          6719                        end;
13904 ;          6720
13905 ;          6721                    end;
13906 ;          6722
13907 ;          6723                end
13908 ;          6724            else                                     !GTR FLG IS NOT SET
13909 ;          6725                begin
13910 ;          6726
13911 ;          6727                if .LOW_CNT eql .HIGH_CNT            !LOW<9:0> SHOULD BE NEQ HIGH<15:10>
13912 ;          6728                then
13913 ;          6729                    begin                            !ERROR IF EQL
```

```
13915 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
13916 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (76)
13917
13918 ;          6730                ERRDF (73, ARR_DAT, DUMPER);
13919 ;          6731                PRINTB (SIX_FMT, WRD_34, PHR_5, WRD_32, WRD_6, WRD_33, WRD_24);
13920 ;          6732                ERR_FLG = ONE;
13921 ;          6733                end;
13922 ;          6734
13923 ;          6735            end
13924 ;          6736
13925 ;          6737        end
13926 ;          6738    else                                    !UNS WAS NOT SET
13927 ;          6739        begin
13928 ;          6740
13929 ;          6741        if .GTR_FLG IS_SET                  !SEE IF GTR FLG IS_SET
13930 ;          6742        then
13931 ;          6743            begin
13932 ;          6744            TEMP = .PROM_DATA;              !LOAD TEMP WITH PROM DATA
13933 ;          6745            TEMP = .TEMP and %o'162000';    !SEE IF THESE BITS ARE SET IN PROM DATA
13934 ;          6746
13935 ;          6747            if .TEMP neq %o'162000'         !IF THESE BITS ARE SET THEN UNS SHOULD BE SET
13936 ;          6748            then
13937 ;          6749                begin                       !ERROR UNS IS NOT SET
13938 ;          6750                ERRDF (74, ARR_DAT, DUMPER);
13939 ;          6751                PRINTB (SIX_FMT, WRD_34, PHR_1, WRD_32, WRD_5, WRD_33, WRD_24);
13940 ;          6752                ERR_FLG = ONE;
13941 ;          6753                end
13942 ;          6754            else                            !BITS 162000 ARE NOT SET
13943 ;          6755                begin
13944 ;          6756
13945 ;          6757                if .LOW_CNT lss .HIGH_CNT   !LOW<9:0> SHOULD BE GEQ HIGH<15:10>
13946 ;          6758                then
13947 ;          6759                    begin                   !ERROR IF LSS
13948 ;          6760                    ERRDF (75, ARR_DAT, DUMPER);
13949 ;          6761                    PRINTB (SIX_FMT, WRD_34, PHR_1, WRD_32, WRD_5, WRD_33, WRD_24);
13950 ;          6762                    ERR_FLG = ONE;
13951 ;          6763                    end;
13952 ;          6764
13953 ;          6765                end
13954 ;          6766
13955 ;          6767            end
13956 ;          6768        else                                !GTR_FLG IS NOT SET
13957 ;          6769            begin
13958 ;          6770
13959 ;          6771            if .LOW_CNT neq .HIGH_CNT       !LOW<9:0> SHOULD EQL HIGH<15:10>
13960 ;          6772            then
13961 ;          6773                begin                       !ERROR IF NEQ
13962 ;          6774                ERRDF (108, ARR_DAT, DUMPER);
13963 ;          6775                PRINTB (SIX_FMT, WRD_34, PHR_1, WRD_32, WRD_5, WRD_33, WRD_24);
13964 ;          6776                ERR_FLG = ONE;
13965 ;          6777                end
13966 ;          6778
13967 ;          6779            end
13968 ;          6780
13969 ;          6781        end;
```

```
 13971 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
 13972 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (76)
 13973
 13974 ;          6782
 13975 ;          6783      if .ERR_FLG IS_SET                 !SEE IF ERROR FLG GOT SET
 13976 ;          6784      then
 13977 ;          6785          begin                          !PRINT FAILING PROM_DATA AND SET DODU_FLG
 13978 ;          6786          PRINTB (FMT_7, .PROM_DATA);
 13979 ;          6787          DODU_FLG = ONE;
 13980 ;          6788          end;
 13981 ;          6789
 13982 ;          6790      ENDSUB;
 13983 ;          6791      end
 13984 ;          6792  until .PROM_DATA eql %o'177777';        !TRY ALL BIT COMBINATIONS
 13985 ;          6793
 13986 ;          6794  if .DODU_FLG IS_SET                     !DROP THIS UNIT IF DODU_FLG IS SET
 13987 ;          6795  then
 13988 ;          6796      begin
 13989 ;          6797      DODU (.ML_LUN);
 13990 ;          6798      DOCLN;
 13991 ;          6799      end;
 13992 ;          6800
 13993 ;          6801  ENDTST;
 13997
 13998                                          .SBTTL  $T26
 14002 047534  004167  134356          $T26:   JSR     R1,$SAVE5                       ;                       6624
 14003 047540  162706  000012                  SUB     #12,SP
 14004 047544  005066  000006                  CLR     6(SP)                           ; DODU.FLG              6666
 14005 047550  012766  177777  000010          MOV     #-1,10(SP)                      ; *,PROM.DATA           6667
 14006 047556  005266  000010          1$:     INC     10(SP)                          ; PROM.DATA             6671
 14007 047562  104402                  2$:     TRAP    2
 14008 047564  152777  000040  143422          BISB    #40,@ML.REG+40                  ;                       6672
 14009 047572  016705  144004                  MOV     ML.DUT,R5
 14010 047576  042705  177770                  BIC     #177770,R5
 14011 047602  142777  000007  143404          BICB    #7,@ML.REG+40
 14012 047610  150577  143400                  BISB    R5,@ML.REG+40
 14013 047614  005001                          CLR     R1                              ; ERR.FLG               6674
 14014 047616  005066  000002                  CLR     2(SP)                           ; LOW.CNT               6675
 14015 047622  005003                          CLR     R3                              ; HIGH.CNT              6676
 14016 047624  005066  000004                  CLR     4(SP)                           ; GTR.FLG               6677
 14017 047630  005005                          CLR     R5                              ; BIT.CNT               6679
 14018 047632  010504                  3$:     MOV     R5,R4                           ; BIT.CNT,*             6681
 14019 047634  006204                          ASR     R4
 14020 047636  006204                          ASR     R4
 14021 047640  006204                          ASR     R4
 14022 047642  012702  000010                  MOV     #10,R2
 14023 047646  060602                          ADD     SP,R2                           ; PROM.DATA,*
 14024 047650  060204                          ADD     R2,R4
 14025 047652  010446                          MOV     R4,-(SP)
```

```
14027                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
14028                                   ;                                                       14-May-1981 13:05:08    PA:<
14029
14030 047654  010546                            MOV     R5,-(SP)                ; BIT.CNT,*
14031 047656  042716  177770                    BIC     #177770,(SP)
14032 047662  012746  000001                    MOV     #1,-(SP)
14033 047666  005046                            CLR     -(SP)
14034 047670  004767  133244                    JSR     PC,BL$GT2
14035 047674  062706  000010                    ADD     #10,SP
14036 047700  005300                            DEC     R0
14037 047702  001002                            BNE     4$
14038 047704  005266  000002                    INC     2(SP)                   ; LOW.CNT
14039 047710  005205                    4$:     INC     R5                      ; BIT.CNT                               6679
14040 047712  020527  000011                    CMP     R5,#11                  ; BIT.CNT,*
14041 047716  003745                            BLE     3$
14042 047720  016603  000010                    MOV     10(SP),R3               ; PROM.DATA,HIGH.CNT                    6683
14043 047724  006203                            ASR     R3                      ; HIGH.CNT
14044 047726  006203                            ASR     R3                      ; HIGH.CNT
14045 047730  000303                            SWAB    R3                      ; HIGH.CNT
14046 047732  042703  177770                    BIC     #177770,R3              ; *,HIGH.CNT
14047 047736  132766  000040  000011            BITB    #40,11(SP)              ; *,PROM.DATA+1                         6685
14048 047744  001401                            BEQ     5$
14049 047746  005203                            INC     R3                      ; HIGH.CNT
14050 047750  132766  000100  000011    5$:     BITB    #100,11(SP)             ; *,PROM.DATA+1                         6687
14051 047756  001401                            BEQ     6$
14052 047760  005203                            INC     R3                      ; HIGH.CNT
14053 047762  005005                    6$:     CLR     R5                      ;                                       6689
14054 047764  105766  000011                    TSTB    11(SP)                  ; PROM.DATA+1
14055 047770  100002                            BPL     7$
14056 047772  005205                            INC     R5
14057 047774  005203                            INC     R3                      ; HIGH.CNT
14058 047776  006005                    7$:     ROR     R5                      ;                                       6691
14059 050000  103003                            BCC     8$
14060 050002  012766  000001  000004            MOV     #1,4(SP)                ; *,GTR.FLG
14061 050010  004767  144146            8$:     JSR     PC,DAT.DM.XFER          ;                                       6693
14062 050014  152777  000100  143252            BISB    #100,@ML.REG+120        ;                                       6694
14063 050022  016605  000010                    MOV     10(SP),R5               ; PROM.DATA,*                           6695
14064 050026  010577  143352                    MOV     R5,@ML.REG+230
14065 050032  012777  000061  143114            MOV     #61,@ML.REG             ;                                       6696
14066 050040  152777  000020  143226            BISB    #20,@ML.REG+120         ;                                       6697
14067 050046  032777  040000  143160            BIT     #40000,@ML.REG+60       ;                                       6699
14068 050054  001500                            BEQ     12$
14069 050056  026627  000004  000001            CMP     4(SP),#1                ; GTR.FLG,*                             6703
14070 050064  001042                            BNE     11$
14071 050066  010516                            MOV     R5,(SP)                 ; *,TEMP                                6706
14072 050070  042716  015777                    BIC     #15777,(SP)             ; *,TEMP                                6707
14073 050074  021627  162000                    CMP     (SP),#-16000            ; TEMP,*                                6709
14074 050100  001003                            BNE     9$
14075 050102  026603  000002                    CMP     2(SP),R3                ; LOW.CNT,HIGH.CNT                      6713
14076 050106  002002                            BGE     10$
14077 050110  000167  000422            9$:     JMP     16$
14078 050114  104455                    10$:    TRAP    55                      ;                                       6716
14079 050116  000110                            .WORD   110
14080 050120  010570                            .WORD   ARR.DAT
14081 050122  024060                            .WORD   DUMPER
```

```
14083                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
14084                                   ;                                                       14-May-1981 13:05:08    PA:<
14085
14086 050124  012746  006552                      MOV     #WRD.24,-(SP)           ;                                       6717
14087 050130  012746  006654                      MOV     #WRD.33,-(SP)
14088 050134  012746  006336                      MOV     #WRD.6,-(SP)
14089 050140  012746  006646                      MOV     #WRD.32,-(SP)
14090 050144  012746  007472                      MOV     #PHR.5,-(SP)
14091 050150  012746  006660                      MOV     #WRD.34,-(SP)
14092 050154  012746  006100                      MOV     #SIX.FMT,-(SP)
14093 050160  012746  000007                      MOV     #7,-(SP)
14094 050164  010600                              MOV     SP,R0                   ; SP,*
14095 050166  104414                              TRAP    14
14096 050170  000556                              BR      15$                     ;                                       6718
14097 050172  026603  000002          11$:        CMP     2(SP),R3                ; LOW.CNT,HIGH.CNT                      6727
14098 050176  001157                              BNE     16$
14099 050200  104455                              TRAP    55                      ;                                       6730
14100 050202  000111                              .WORD   111
14101 050204  010570                              .WORD   ARR.DAT
14102 050206  024060                              .WORD   DUMPER
14103 050210  012746  006552                      MOV     #WRD.24,-(SP)           ;                                       6731
14104 050214  012746  006654                      MOV     #WRD.33,-(SP)
14105 050220  012746  006336                      MOV     #WRD.6,-(SP)
14106 050224  012746  006646                      MOV     #WRD.32,-(SP)
14107 050230  012746  007472                      MOV     #PHR.5,-(SP)
14108 050234  012746  006660                      MOV     #WRD.34,-(SP)
14109 050240  012746  006100                      MOV     #SIX.FMT,-(SP)
14110 050244  012746  000007                      MOV     #7,-(SP)
14111 050250  010600                              MOV     SP,R0                   ; SP,*
14112 050252  104414                              TRAP    14
14113 050254  000524                              BR      15$                     ;                                       6732
14114 050256  026627  000004  000001  12$:        CMP     4(SP),#1                ; GTR.FLG,*                             6741
14115 050264  001067                              BNE     14$
14116 050266  010516                              MOV     R5,(SP)                 ; *,TEMP                                6744
14117 050270  042716  015777                      BIC     #15777,(SP)             ; *,TEMP                                6745
14118 050274  021627  162000                      CMP     (SP),#-16000            ; TEMP,*                                6747
14119 050300  001427                              BEQ     13$
14120 050302  104455                              TRAP    55                      ;                                       6750
14121 050304  000112                              .WORD   112
14122 050306  010570                              .WORD   ARR.DAT
14123 050310  024060                              .WORD   DUMPER
14124 050312  012746  006552                      MOV     #WRD.24,-(SP)           ;                                       6751
14125 050316  012746  006654                      MOV     #WRD.33,-(SP)
14126 050322  012746  006330                      MOV     #WRD.5,-(SP)
14127 050326  012746  006646                      MOV     #WRD.32,-(SP)
14128 050332  012746  007366                      MOV     #PHR.1,-(SP)
14129 050336  012746  006660                      MOV     #WRD.34,-(SP)
14130 050342  012746  006100                      MOV     #SIX.FMT,-(SP)
14131 050346  012746  000007                      MOV     #7,-(SP)
14132 050352  010600                              MOV     SP,R0                   ; SP,*
14133 050354  104414                              TRAP    14
14134 050356  000463                              BR      15$                     ;                                       6752
14135 050360  026603  000002          13$:        CMP     2(SP),R3                ; LOW.CNT,HIGH.CNT                      6757
14136 050364  002064                              BGE     16$
14137 050366  104455                              TRAP    55                      ;                                       6760
```

```
14139                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
14140                                   ;                                                       14-May-1981 13:05:08    PA:<
14141
14142 050370  000113                            .WORD   113
14143 050372  010570                            .WORD   ARR.DAT
14144 050374  024060                            .WORD   DUMPER
14145 050376  012746  006552                    MOV     #WRD.24,-(SP)           ;                               6761
14146 050402  012746  006654                    MOV     #WRD.33,-(SP)
14147 050406  012746  006330                    MOV     #WRD.5,-(SP)
14148 050412  012746  006646                    MOV     #WRD.32,-(SP)
14149 050416  012746  007366                    MOV     #PHR.1,-(SP)
14150 050422  012746  006660                    MOV     #WRD.34,-(SP)
14151 050426  012746  006100                    MOV     #SIX.FMT,-(SP)
14152 050432  012746  000007                    MOV     #7,-(SP)
14153 050436  010600                            MOV     SP,R0                   ; SP,*
14154 050440  104414                            TRAP    14
14155 050442  000431                            BR      15$                     ;                               6762
14156 050444  026603  000002          14$:      CMP     2(SP),R3                ; LOW.CNT,HIGH.CNT              6771
14157 050450  001432                            BEQ     16$
14158 050452  104455                            TRAP    55                      ;                               6774
14159 050454  000154                            .WORD   154
14160 050456  010570                            .WORD   ARR.DAT
14161 050460  024060                            .WORD   DUMPER
14162 050462  012746  006552                    MOV     #WRD.24,-(SP)           ;                               6775
14163 050466  012746  006654                    MOV     #WRD.33,-(SP)
14164 050472  012746  006330                    MOV     #WRD.5,-(SP)
14165 050476  012746  006646                    MOV     #WRD.32,-(SP)
14166 050502  012746  007366                    MOV     #PHR.1,-(SP)
14167 050506  012746  006660                    MOV     #WRD.34,-(SP)
14168 050512  012746  006100                    MOV     #SIX.FMT,-(SP)
14169 050516  012746  000007                    MOV     #7,-(SP)
14170 050522  010600                            MOV     SP,R0                   ; SP,*
14171 050524  104414                            TRAP    14
14172 050526  012701  000001          15$:      MOV     #1,R1                   ; *,ERR.FLG                     6776
14173 050532  062706  000020                    ADD     #20,SP                  ;                               6773
14174 050536  020127  000001          16$:      CMP     R1,#1                   ; ERR.FLG,*                     6783
14175 050542  001014                            BNE     17$
14176 050544  010546                            MOV     R5,-(SP)                ;                               6786
14177 050546  012746  004562                    MOV     #FMT.7,-(SP)
14178 050552  012746  000002                    MOV     #2,-(SP)
14179 050556  010600                            MOV     SP,R0                   ; SP,*
14180 050560  104414                            TRAP    14
14181 050562  012766  000001  000014            MOV     #1,14(SP)               ; *,DODU.FLG                    6787
14182 050570  062706  000006                    ADD     #6,SP                   ;                               6785
14183 050574  104467                  17$:      TRAP    67                      ;                               6788
14184 050576  006000                            ROR     R0
14185 050600  103002                            BHIS    18$
14186 050602  000167  176754                    JMP     2$
14187 050606  005205                  18$:      INC     R5                      ;                               6792
14188 050610  001402                            BEQ     19$
14189 050612  000167  176740                    JMP     1$
14190 050616  026627  000006  000001  19$:      CMP     6(SP),#1                ; DODU.FLG,*                    6794
14191 050624  001004                            BNE     20$
14192 050626  016700  142746                    MOV     ML.LUN,R0               ;                               6797
14193 050632  104451                            TRAP    51
```

```
                                ;ML4                                                    14-May-1981 13:48:54    TOPS
                                ;                                                       14-May-1981 13:05:08    PA:<

050634  104444                          TRAP    44
050636  062706  000012          20$:    ADD     #12,SP                  ;                                       6624
050642  000207                          RTS     PC

                                ; Routine Size:  292 words
                                ; Maximum stack depth per invocation:  19 words




                                        .SBTTL  T26
050644                          T26::
050644  004767  176664          1$:     JSR     PC,$T26                 ;                                       6799
050650  104466                          TRAP    66
050652  006000                          ROR     R0
050654  103773                          BLO     1$
050656  000207                          RTS     PC

                                ; Routine Size:  6 words
                                ; Maximum stack depth per invocation:  0 words


;          6802  !<BLF/PAGE>
```

```
14235 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
14236 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (77)
14237
14238 ;          6803  !
14239 ;          6804  !
14240 ;          6805  BGNTST;
14241 ;          6806
14242 ;          6807  !++
14243 ;          6808  ! TEST NUMBER:  TST 27
14244 ;          6809  !
14245 ;          6810  ! TEST NAME:    INITIAL ARRAY TEST
14246 ;          6811  !
14247 ;          6812  ! TEST DESCRIPTION:
14248 ;          6813  !
14249 ;          6814  !       DUE TO THE NATURE OF THE DEVICE
14250 ;          6815  !       THERE EXISTS KNOWN BAD ARRAY
14251 ;          6816  !       DATA LOCATIONS.
14252 ;          6817  !
14253 ;          6818  !       THEREFORE TO INITIALLY TEST THE ARRAYS'
14254 ;          6819  !       TIMING AND CONTROL LOGIC A BAD
14255 ;          6820  !       NIBBLE THRESHOLD OF 36 BAD NIBBLES
14256 ;          6821  !       OUT OF 100 NIBBLES TESTED WILL BE
14257 ;          6822  !       TOLERATED BEFORE DETERMINING CONTROL
14258 ;          6823  !       LOGIC TO BE IN ERROR.
14259 ;          6824  !
14260 ;          6825  !       THE ARRAYS' ARE INITIALLY TESTED BY:
14261 ;          6826  !
14262 ;          6827  !       1.  VIA DAT_DM MOD WRITE DATA PATTERNS
14263 ;          6828  !           OF 1'S AND 0'S TO 5 ARRAY
14264 ;          6829  !           WORDS.
14265 ;          6830  !
14266 ;          6831  !       2.  TEST EACH NIBBLE (4 BITS) FOR
14267 ;          6832  !           1'S AND 0'S AND COUNT EACH BAD
14268 ;          6833  !           NIBBLE ENCOUNTERED.
14269 ;          6834  !
14270 ;          6835  !       3.  IF ACCUMULATED BAD NIBBLES
14271 ;          6836  !           EXCEED 36 THEN REPORT AN ERROR.
14272 ;          6837  !
14273 ;          6838  !--
14274 ;          6839
14275 ;          6840  local
14276 ;          6841      TST_PAT,                          !TEST PATTERN
14277 ;          6842      BAD_NIB_CNT,                      !NUMBER OF BAD NIBBLES FOUND
14278 ;          6843      ERR_FLG;                          !ERROR FLAG
14279 ;          6844
14280 ;          6845  TST_PAT = ONES;
14281 ;          6846  BAD_NIB_CNT = ZEROES;
14282 ;          6847
14283 ;          6848  incr TWICE from 0 to 1 do             !REPEAT LOOP TWICE
14284 ;          6849      begin
14285 ;          6850      BGNSUB;
14286 ;          6851      CLR_MBUS;
14287 ;          6852      MLDT = .TST_PAT;                  !LOAD TEST PATTERN INTO DIAG REGISTERS
14288 ;          6853      MLD2 = .TST_PAT;
14289 ;          6854      MLE2 = .TST_PAT;
```

```
14291 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
14292 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (77)
14293
14294 ;         6855      DAT_DM = ONE;                       !SET DATA DIAG MODE
14295 ;         6856      FIRST_BLK_XFER ();                  !SET UP A FIRST BLK XFERR
14296 ;         6857      MLCS1 = write;                      !DO A MBUS WRITE FUNCTION
14297 ;         6858
14298 ;         6859      incr CNT from 0 to 4 do             !CLOCK 5 WORDS INTO MEMORY
14299 ;         6860          begin
14300 ;         6861          DELAY (ONE_US);
14301 ;         6862          DAT_CLK = ONE;
14302 ;         6863          end;
14303 ;         6864
14304 ;         6865      CLR_MBUS;
14305 ;         6866      DAT_DM = ONE;                       !SET DATA DIAG MODE
14306 ;         6867      FIRST_BLK_XFER ();                  !SET UP A FIRST BLK XFERR
14307 ;         6868      MLCS1 = read;                       !DO A READ FUNCTION
14308 ;         6869      DELAY (ONE_US);
14309 ;         6870
14310 ;         6871      incr ARR_WRD from 0 to 4 do         !READ THE 5 WORD IN MEMORY
14311 ;         6872          begin
14312 ;         6873          DAT_CLK = ONE;                  !CLOCK OUT A WORD INTO THE DIAG REGS
14313 ;         6874          DELAY (ONE_US);
14314 ;         6875          RD_LNG_WRD;                     !READ THE DIAG REGISTERS
14315 ;         6876
14316 ;         6877          incr NIB_PTR from 0 to 9 do     !READ THE 10 NIBBLES IN THE WORD
14317 ;         6878              begin
14318 ;         6879              TST_LNG_WRD (.NIB_PTR, .TST_PAT, ERR_FLG);  !COMPARE TST PAT TO NIBBLE UNDER TEST
14319 ;         6880
14320 ;         6881              if .ERR_FLG IS_SET then BAD_NIB_CNT = .BAD_NIB_CNT + 1;
14321 ;         6882
14322 ;         6883                                          !INCREMENT BAD_NIBBLE COUNT IFERR_FLG SET
14323 ;         6884              end;
14324 ;         6885
14325 ;         6886          end;
14326 ;         6887
14327 ;         6888      TST_PAT = not .TST_PAT;             !REPEAT WITH COMPLIMENT TST PAT
14328 ;         6889      ENDSUB;
14329 ;         6890      end;
14330 ;         6891
14331 ;         6892  if .BAD_NIB_CNT gtr 36                  !SEE IF 36 OUT OF THE 100 XFERRED WHERE BAD
14332 ;         6893  then
14333 ;         6894      begin                               !ERROR IF GTR 36
14334 ;         6895      ERRDF (77, ASYNC, DUMPER);
14335 ;         6896      PRINTB (FIV_FMT, WRD_22, PHR_4, WRD_12, WRD_45, FNC_14);
14336 ;         6897      DODU (.ML_LUN);
14337 ;         6898      DOCLN;
14338 ;         6899      end;
14339 ;         6900
14340 ;         6901  ENDTST;
14344
14345                                   .SBTTL  ST27
```

```
14347                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
14348                                   ;                                                       14-May-1981 13:05:08    PA:<
14349
14353 050660  004167  133232            $T27:   JSR     R1,$SAVE5               ;                                       6801
14354 050664  024646                            CMP     -(SP),-(SP)
14355 050666  012701  177777                    MOV     #-1,R1                  ; *,TST.PAT                             6845
14356 050672  005046                            CLR     -(SP)                   ; BAD.NIB.CNT                           6846
14357 050674  005005                            CLR     R5                      ; TWICE                                 6848
14358 050676  104402                    1$:     TRAP    2                       ;                                       6849
14359 050700  152777  000040  142306            BISB    #40,@ML.REG+40          ;                                       6850
14360 050706  016704  142670                    MOV     ML.DUT,R4
14361 050712  042704  177770                    BIC     #177770,R4
14362 050716  142777  000007  142270            BICB    #7,@ML.REG+40
14363 050724  150477  142264                    BISB    R4,@ML.REG+40
14364 050730  010177  142410                    MOV     R1,@ML.REG+170          ; TST.PAT,*                             6852
14365 050734  010177  142414                    MOV     R1,@ML.REG+200          ; TST.PAT,*                             6853
14366 050740  010177  142370                    MOV     R1,@ML.REG+160          ; TST.PAT,*                             6854
14367 050744  152777  000010  142322            BISB    #10,@ML.REG+120         ;                                       6855
14368 050752  004767  145062                    JSR     PC,FIRST.BLK.XFER       ;                                       6856
14369 050756  012777  000061  142170            MOV     #61,@ML.REG             ;                                       6857
14370 050764  005002                            CLR     R2                      ; CNT                                   6859
14371 050766  012703  000001            2$:     MOV     #1,R3                   ; *,$$TMP2                              6861
14372 050772  001411                    3$:     BEQ     6$
14373 050774  016704  131116                    MOV     L$DLY,R4                ; *,$$TMP1
14374 051000  001404                            BEQ     5$
14375 051002  005066  000004            4$:     CLR     4(SP)                   ; $$TMP
14376 051006  005304                            DEC     R4                      ; $$TMP1
14377 051010  001374                            BNE     4$
14378 051012  005303                    5$:     DEC     R3                      ; $$TMP2
14379 051014  000766                            BR      3$
14380 051016  152777  000020  142250    6$:     BISB    #20,@ML.REG+120         ;                                       6862
14381 051024  005202                            INC     R2                      ; CNT                                   6859
14382 051026  020227  000004                    CMP     R2,#4                   ; CNT,*
14383 051032  003755                            BLE     2$
14384 051034  152777  000040  142152            BISB    #40,@ML.REG+40          ;                                       6863
14385 051042  016704  142534                    MOV     ML.DUT,R4
14386 051046  042704  177770                    BIC     #177770,R4
14387 051052  142777  000007  142134            BICB    #7,@ML.REG+40
14388 051060  150477  142130                    BISB    R4,@ML.REG+40
14389 051064  152777  000010  142202            BISB    #10,@ML.REG+120         ;                                       6866
14390 051072  004767  144742                    JSR     PC,FIRST.BLK.XFER       ;                                       6867
14391 051076  012777  000071  142050            MOV     #71,@ML.REG             ;                                       6868
14392 051104  012703  000001                    MOV     #1,R3                   ; *,$$TMP2                              6869
14393 051110  001411                    7$:     BEQ     10$
14394 051112  016704  131000                    MOV     L$DLY,R4                ; *,$$TMP1
14395 051116  001404                            BEQ     9$
14396 051120  005066  000004            8$:     CLR     4(SP)                   ; $$TMP
14397 051124  005304                            DEC     R4                      ; $$TMP1
14398 051126  001374                            BNE     8$
14399 051130  005303                    9$:     DEC     R3                      ; $$TMP2
14400 051132  000766                            BR      7$
14401 051134  005002                    10$:    CLR     R2                      ; ARR.WRD                               6871
```

```
14403                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
14404                                   ;                                                       14-May-1981 13:05:08    PA:<
14405
14406 051136  152777  000020  142130  11$:    BISB    #20,@ML.REG+120         ;                                       6873
14407 051144  012703  000001                  MOV     #1,R3                   ; *,$$TMP2                              6874
14408 051150  001411                  12$:    BEQ     15$
14409 051152  016704  130740                  MOV     L$DLY,R4                ; *,$$TMP1
14410 051156  001404                          BEQ     14$
14411 051160  005066  000004          13$:    CLR     4(SP)                   ; $$TMP
14412 051164  005304                          DEC     R4                      ; $$TMP1
14413 051166  001374                          BNE     13$
14414 051170  005303                  14$:    DEC     R3                      ; $$TMP2
14415 051172  000766                          BR      12$
14416 051174  017767  142144  137750  15$:    MOV     @ML.REG+170,D1.TEMP
14417 051202  017767  142146  137744          MOV     @ML.REG+200,D2.TEMP
14418 051210  017767  142120  137740          MOV     @ML.REG+160,E2.TEMP
14419 051216  005004                          CLR     R4                      ; NIB.PTR                               6877
14420 051220  010446                  16$:    MOV     R4,-(SP)                ; NIB.PTR,*                             6879
14421 051222  010146                          MOV     R1,-(SP)                ; TST.PAT,*
14422 051224  012746  000010                  MOV     #10,-(SP)
14423 051230  060616                          ADD     SP,(SP)                 ; ERR.FLG,*
14424 051232  004767  144716                  JSR     PC,TST.LNG.WRD
14425 051236  026627  000010  000001          CMP     10(SP),#1               ; ERR.FLG,*                             6881
14426 051244  001002                          BNE     17$
14427 051246  005266  000006                  INC     6(SP)                   ; BAD.NIB.CNT
14428 051252  062706  000006          17$:    ADD     #6,SP                   ;                                       6878
14429 051256  005204                          INC     R4                      ; NIB.PTR                               6877
14430 051260  020427  000011                  CMP     R4,#11                  ; NIB.PTR,*
14431 051264  003755                          BLE     16$
14432 051266  005202                          INC     R2                      ; ARR.WRD                               6871
14433 051270  020227  000004                  CMP     R2,#4                   ; ARR.WRD,*
14434 051274  003720                          BLE     11$
14435 051276  005101                          COM     R1                      ; TST.PAT                               6888
14436 051300  104467                          TRAP    67
14437 051302  006000                          ROR     R0
14438 051304  103002                          BHIS    19$
14439 051306  000167  177364          18$:    JMP     1$
14440 051312  005205                  19$:    INC     R5                      ; TWICE                                 6848
14441 051314  020527  000001                  CMP     R5,#1                   ; TWICE,*
14442 051320  003772                          BLE     18$
14443 051322  021627  000044                  CMP     (SP),#44                ; BAD.NIB.CNT,*                         6892
14444 051326  003432                          BLE     20$
14445 051330  104455                          TRAP    55                      ;                                       6895
14446 051332  000115                          .WORD   115
14447 051334  010464                          .WORD   ASYNC
14448 051336  024060                          .WORD   DUMPER
14449 051340  012746  010122                  MOV     #FNC.14,-(SP)           ;                                       6896
14450 051344  012746  006772                  MOV     #WRD.45,-(SP)
14451 051350  012746  006426                  MOV     #WRD.12,-(SP)
14452 051354  012746  007454                  MOV     #PHR.4,-(SP)
14453 051360  012746  006530                  MOV     #WRD.22,-(SP)
14454 051364  012746  006062                  MOV     #FIV.FMT,-(SP)
14455 051370  012746  000006                  MOV     #6,-(SP)
14456 051374  010600                          MOV     SP,R0                   ; SP,*
14457 051376  104414                          TRAP    14
```

```
14459                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
14460                                   ;                                                       14-May-1981 13:05:08    PA:<
14461
14462 051400  016700  142174                    MOV     ML.LUN,R0                       ;                               6897
14463 051404  104451                            TRAP    51
14464 051406  104444                            TRAP    44
14465 051410  062706  000016                    ADD     #16,SP                          ;                               6894
14466 051414  062706  000006            20$:    ADD     #6,SP                           ;                               6801
14467 051420  000207                            RTS     PC
14468
14469                                   ; Routine Size:  177 words
14470                                   ; Maximum stack depth per invocation:  16 words
14475
14476
14480
14481                                           .SBTTL  T27
14485 051422                            T27::
14486 051422  004767  177232            1$:     JSR     PC,$T27                         ;                               6899
14487 051426  104466                            TRAP    66
14488 051430  006000                            ROR     R0
14489 051432  103773                            BLO     1$
14490 051434  000207                            RTS     PC
14491
14492                                   ; Routine Size:  6 words
14493                                   ; Maximum stack depth per invocation:  0 words
14498
14499
14500 ;          6902  !<BLF/PAGE>
```

```
14502 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
14503 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (78)
14504
14505 ;          6903  !
14506 ;          6904  BGNTST;
14507 ;          6905
14508 ;          6906  !++
14509 ;          6907  ! TEST NUMBER:  TST 28
14510 ;          6908  !
14511 ;          6909  ! TEST NAME:    PROM SELECTION TEST
14512 ;          6910  !
14513 ;          6911  ! TEST DESCRIPTION:
14514 ;          6912  !       DUE TO THE NATURE OF THE DEVICE
14515 ;          6913  !       AND OF THE ARRAY MODULES' UV
14516 ;          6914  !       PROMS, ONLY PROM READS ARE
14517 ;          6915  !       ALLOWED DURING DIAG TESTING.
14518 ;          6916  !
14519 ;          6917  !       THEREFORE THE ARRAY MODULE UV PROMS
14520 ;          6918  !       ARE TESTED FOR UNIQUE SELECTION BY:
14521 ;          6919  !
14522 ;          6920  !       1.  AT EACH PRESENT ARRAY MODULE WRITE 127 ARRAY WORDS WITH 1'S/0'S PATTERN.
14523 ;          6921  !
14524 ;          6922  !       2.  READ THE UV PROMS AT THEIR RESPECTIVE ARRAY WORD LOCATION AND SEE IF
14525 ;          6923  !           THE PROMS MASK BAD NIBBLE LOCATIONS (ENCOUNTERED BAD NIBBLES
14526 ;          6924  !           INDICATES INCORRECT MASKING).  COUNT EACH BAD NIBBLE ENCOUNTERED AT AN
14527 ;          6925  !           ARRAY MODULE.
14528 ;          6926  !
14529 ;          6927  !       3.  ALLOW A THRESHOLD OF 5 BAD NIBBLES AT ANY ARRAY MODULE.
14530 ;          6928  !
14531 ;          6929  !       4.  REPORT PROM SEL ERRORS AT RESPECTIVE ARRAY MODULE IF THE
14532 ;          6930  !           THRESHOLD IS EXCEEDED.
14533 ;          6931  !
14534 ;          6932  ! IMPLICIT INPUTS:
14535 ;          6933  !       PD_TEMP:
14536 ;          6934  !       A BIT VECTOR OF 16 BITS WHERE
14537 ;          6935  !       THE READ PROM DATA IS STORED
14538 ;          6936  !       AND ACCESSED FROM.
14539 ;          6937  !
14540 ;          6938  !       IO_BUF
14541 ;          6939  !       A VECTOR OF 256 WORDS WHERE
14542 ;          6940  !       DATA FOR MBUS READS AND WRITE
14543 ;          6941  !       FUNCTION ARE FOUND.
14544 ;          6942  !
14545 ;          6943  !--
14546 ;          6944
14547 ;          6945  local
14548 ;          6946      DODU_FLG,                                   !DROP UNIT FLG
14549 ;          6947      ERR_FLG,                                    !ERROR FLG
14550 ;          6948      TST_PAT,                                    !TEST PATTERN
14551 ;          6949      ERR_CNT;                                    !ERROR COUNT
14552 ;          6950
14553 ;          6951  DODU_FLG = ZERO;
14554 ;          6952  TST_PAT = ONES;
14555 ;          6953
14556 ;          6954  incr ARR_SEL from 0 to .LST_ARR by .ARR_INC do  !TEST ALL PRESENT ARRAYS
```

```
14558 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
14559 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (78)
14560
14561 ;          6955       begin
14562 ;          6956       ERR_CNT = ZERO;
14563 ;          6957       BGNSUB;
14564 ;          6958
14565 ;          6959       incr TWICE from 0 to 1 do                !REPEAT LOOP TWICE
14566 ;          6960           begin
14567 ;          6961           CLR_MBUS;
14568 ;          6962           MLDT = .TST_PAT;                     !LOAD DATA DIAG REGISTERS WITH TST_PAT
14569 ;          6963           MLD2 = .TST_PAT;
14570 ;          6964           MLE2 = .TST_PAT;
14571 ;          6965           DAT_DM = ONE;                        !SET DATA DIAG MODE
14572 ;          6966           MLWC = not 255;                      !LOAD WORD COUNT
14573 ;          6967           MLBA = IO_BUF;                       !LOAD UBUS ADRS
14574 ;          6968           MLDA = .ARR_SEL;                     !LOAD SECTOR
14575 ;          6969           MLCS1 = write;                       !DO A WRITE FUNCTION
14576 ;          6970
14577 ;          6971           incr CNT from 0 to 127 do            !CLOCK IN 127 WORDS
14578 ;          6972               begin
14579 ;          6973               DELAY (ONE_US);
14580 ;          6974               DAT_CLK = ONE;
14581 ;          6975               end;
14582 ;          6976
14583 ;          6977           CLR_MBUS;
14584 ;          6978           DAT_DM = ONE;                        !SET DATA DIAG MODE
14585 ;          6979           MLWC = not 255;                      !LOAD WORD COUNT
14586 ;          6980           MLBA = IO_BUF;                       !LOAD UBUS ADRS
14587 ;          6981           MLDA = .ARR_SEL;                     !LOAD SECTOR
14588 ;          6982           MLCS1 = read;                        !DO A READ FUNCTION
14589 ;          6983           DELAY (ONE_US);
14590 ;          6984
14591 ;          6985           incr WD_CNT from 0 to 127 do         !READ THE 10 WORDS
14592 ;          6986               begin
14593 ;          6987               PD_TEMP = .MLPD;                 !GET PROM DATA FOR THIS WORD
14594 ;          6988               DAT_CLK = ONE;                   !CLOCK THIS WORD INTO DIAG REG
14595 ;          6989               DELAY (ONE_US);
14596 ;          6990               RD_LNG_WRD;                      !READ DIAG REG FOR THIS WORD
14597 ;          6991
14598 ;          6992               incr NIB_PTR from 0 to 9 do      !LOOK AT ALL 10 NIBBLE
14599 ;          6993
14600 ;          6994                   if .PD_TEMP [.NIB_PTR] IS_NOT_SET       !FIND GOOD NIBBLES
14601 ;          6995                   then
14602 ;          6996                       begin
14603 ;          6997                       TST_LNG_WRD (.NIB_PTR, .TST_PAT, ERR_FLG);  !COMPARE NIBBLE TO TST_PAT
14604 ;          6998
14605 ;          6999                       if .ERR_FLG IS_SET then ERR_CNT = .ERR_CNT + 1;
14606 ;          7000
14607 ;          7001                                                !INCREMENT ERROR COUNT IF ERROR FLG IS SET
14608 ;          7002                       end;
14609 ;          7003
14610 ;          7004               end;
14611 ;          7005
14612 ;          7006       TST_PAT = not .TST_PAT;                  !REPEAT WITH COMPLIMENT DATA
```

```
  14614 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
  14615 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (78)
  14616
  14617 ;          7007          end;
  14618 ;          7008
  14619 ;          7009      if .ERR_CNT gtr 5                  !ALLOW 5 ERROR BEFORE ERRORING
  14620 ;          7010      then
  14621 ;          7011          begin                          !ERROR IF GTR 5
  14622 ;          7012          ERRDF (78, ARR_DAT, DUMPER);
  14623 ;          7013          PRINTB (THR_FMT, WRD_35, WRD_37, WRD_10);
  14624 ;          7014          PRINTB (FMT_9, .ARR_SEL);
  14625 ;          7015          DODU_FLG = ONE;
  14626 ;          7016          end;
  14627 ;          7017
  14628 ;          7018      ENDSUB;
  14629 ;          7019
  14630 ;          7020      if .DODU_FLG IS_SET                !DROP THIS UNIT IF DODU_FLG IS SET
  14631 ;          7021      then
  14632 ;          7022          begin
  14633 ;          7023          DODU (.ML_LUN);
  14634 ;          7024          DOCLN;
  14635 ;          7025          end;
  14636 ;          7026
  14637 ;          7027      end;
  14638 ;          7028
  14639 ;          7029  ENDTST;
  14643
  14644                                    .SBTTL  $T28
  14648 051436  004167  132454        $T28: JSR     R1,$SAVE5                    ;                          6901
  14649 051442  162706  000016              SUB     #16,SP
  14650 051446  005066  000010              CLR     10(SP)                       ; DODU.FLG                 6951
  14651 051452  012702  177777              MOV     #-1,R2                       ; *,TST.PAT                6952
  14652 051456  016766  140114  000006      MOV     LST.ARR,6(SP)                ;                          6954
  14653 051464  016766  140072  000004      MOV     ARR.INC,4(SP)
  14654 051472  005001                      CLR     R1                           ; ARR.SEL
  14655 051474  000167  000666              JMP     25$
  14656 051500  005066  000002        1$:   CLR     2(SP)                        ; ERR.CNT                  6956
  14657 051504  104402                2$:   TRAP    2
  14658 051506  005016                      CLR     (SP)                         ; TWICE                    6959
  14659 051510  152777  000040  141476  3$: BISB    #40,@ML.REG+40               ;                          6960
  14660 051516  016705  142060              MOV     ML.DUT,R5
  14661 051522  042705  177770              BIC     #177770,R5
  14662 051526  142777  000007  141460      BICB    #7,@ML.REG+40
  14663 051534  150577  141454              BISB    R5,@ML.REG+40
  14664 051540  010277  141600              MOV     R2,@ML.REG+170               ; TST.PAT,*                6962
  14665 051544  010277  141604              MOV     R2,@ML.REG+200               ; TST.PAT,*                6963
  14666 051550  010277  141560              MOV     R2,@ML.REG+160               ; TST.PAT,*                6964
  14667 051554  152777  000010  141512      BISB    #10,@ML.REG+120              ;                          6965
  14668 051562  012777  177400  141374      MOV     #-400,@ML.REG+10             ;                          6966
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                                                                                                        6967
051570  012777  011600  141376          MOV     #IO.BUF,@ML.REG+20      ;                                             6968
051576  010177  141402                  MOV     R1,@ML.REG+30           ; ARR.SEL,*                                   6969
051602  012777  000061  141344          MOV     #61,@ML.REG             ;                                             6971
051610  005003                          CLR     R3                      ; CNT                                         6973
051612  012704  000001          4$:     MOV     #1,R4                   ; *,$$TMP2
051616  001411                  5$:     BEQ     8$
051620  016705  130272                  MOV     L$DLY,R5                ; *,$$TMP1
051624  001404                          BEQ     7$
051626  005066  000014          6$:     CLR     14(SP)                  ; $$TMP
051632  005305                          DEC     R5                      ; $$TMP1
051634  001374                          BNE     6$
051636  005304                  7$:     DEC     R4                      ; $$TMP2
051640  000766                          BR      5$
051642  152777  000020  141424  8$:     BISB    #20,@ML.REG+120         ;                                             6974
051650  005203                          INC     R3                      ; CNT                                         6971
051652  020327  000177                  CMP     R3,#177                 ; CNT,*
051656  003755                          BLE     4$
051660  152777  000040  141326          BISB    #40,@ML.REG+40          ;                                             6975
051666  016705  141710                  MOV     ML.DUT,R5
051672  042705  177770                  BIC     #177770,R5
051676  142777  000007  141310          BICB    #7,@ML.REG+40
051704  150577  141304                  BISB    R5,@ML.REG+40
051710  152777  000010  141356          BISB    #10,@ML.REG+120         ;                                             6978
051716  012777  177400  141240          MOV     #-400,@ML.REG+10        ;                                             6979
051724  012777  011600  141242          MOV     #IO.BUF,@ML.REG+20      ;                                             6980
051732  010177  141246                  MOV     R1,@ML.REG+30           ; ARR.SEL,*                                   6981
051736  012777  000071  141210          MOV     #71,@ML.REG             ;                                             6982
051744  012704  000001                  MOV     #1,R4                   ; *,$$TMP2                                    6983
051750  001411                  9$:     BEQ     12$
051752  016705  130140                  MOV     L$DLY,R5                ; *,$$TMP1
051756  001404                          BEQ     11$
051760  005066  000014          10$:    CLR     14(SP)                  ; $$TMP
051764  005305                          DEC     R5                      ; $$TMP1
051766  001374                          BNE     10$
051770  005304                  11$:    DEC     R4                      ; $$TMP2
051772  000766                          BR      9$
051774  005003                  12$:    CLR     R3                      ; WD.CNT                                      6985
051776  017767  141402  141114  13$:    MOV     @ML.REG+230,PD.TEMP     ;                                             6987
052004  152777  000020  141262          BISB    #20,@ML.REG+120         ;                                             6988
052012  012704  000001                  MOV     #1,R4                   ; *,$$TMP2                                    6989
052016  001411                  14$:    BEQ     17$
052020  016705  130072                  MOV     L$DLY,R5                ; *,$$TMP1
052024  001404                          BEQ     16$
052026  005066  000014          15$:    CLR     14(SP)                  ; $$TMP
052032  005305                          DEC     R5                      ; $$TMP1
052034  001374                          BNE     15$
052036  005304                  16$:    DEC     R4                      ; $$TMP2
052040  000766                          BR      14$
052042  017767  141276  137102  17$:    MOV     @ML.REG+170,D1.TEMP
052050  017767  141300  137076          MOV     @ML.REG+200,D2.TEMP
052056  017767  141252  137072          MOV     @ML.REG+160,E2.TEMP
052064  005004                          CLR     R4                      ; NIB.PTR                                     6992
```

```
14726                          ;ML4                                                    14-May-1981 13:48:54    TOPS
14727                          ;                                                       14-May-1981 13:05:08    PA:<
14728
14729 052066  010405           18$:    MOV     R4,R5                   ; NIB.PTR,*                             6994
14730 052070  006205                   ASR     R5
14731 052072  006205                   ASR     R5
14732 052074  006205                   ASR     R5
14733 052076  062705  013120           ADD     #PD.TEMP,R5
14734 052102  010546                   MOV     R5,-(SP)
14735 052104  010446                   MOV     R4,-(SP)                ; NIB.PTR,*
14736 052106  042716  177770           BIC     #177770,(SP)
14737 052112  012746  000001           MOV     #1,-(SP)
14738 052116  005046                   CLR     -(SP)
14739 052120  004767  131014           JSR     PC,BL$GT2
14740 052124  062706  000010           ADD     #10,SP
14741 052130  005700                   TST     R0
14742 052132  001017                   BNE     20$
14743 052134  010446                   MOV     R4,-(SP)                ; NIB.PTR,*                             6997
14744 052136  010246                   MOV     R2,-(SP)                ; TST.PAT,*
14745 052140  012746  000020           MOV     #20,-(SP)
14746 052144  060616                   ADD     SP,(SP)                 ; ERR.FLG,*
14747 052146  004767  144002           JSR     PC,TST.LNG.WRD
14748 052152  026627  000020  000001   CMP     20(SP),#1               ; ERR.FLG,*                             6999
14749 052160  001002                   BNE     19$
14750 052162  005266  000010           INC     10(SP)                  ; ERR.CNT
14751 052166  062706  000006   19$:    ADD     #6,SP                   ;                                       6996
14752 052172  005204           20$:    INC     R4                      ; NIB.PTR                               6992
14753 052174  020427  000011           CMP     R4,#11                  ; NIB.PTR,*
14754 052200  003732                   BLE     18$
14755 052202  005203                   INC     R3                      ; WD.CNT                                6985
14756 052204  020327  000177           CMP     R3,#177                 ; WD.CNT,*
14757 052210  003672                   BLE     13$
14758 052212  005102                   COM     R2                      ; TST.PAT                               7006
14759 052214  005216                   INC     (SP)                    ; TWICE                                 6959
14760 052216  021627  000001           CMP     (SP),#1                 ; TWICE,*
14761 052222  003002                   BGT     21$
14762 052224  000167  177260           JMP     3$
14763 052230  026627  000002  000005  21$:    CMP     2(SP),#5                ; ERR.CNT,*                     7009
14764 052236  003434                   BLE     22$
14765 052240  104455                   TRAP    55                      ;                                       7012
14766 052242  000116                   .WORD   116
14767 052244  010570                   .WORD   ARR.DAT
14768 052246  024060                   .WORD   DUMPER
14769 052250  012746  006406           MOV     #WRD.10,-(SP)           ;                                       7013
14770 052254  012746  006700           MOV     #WRD.37,-(SP)
14771 052260  012746  006666           MOV     #WRD.35,-(SP)
14772 052264  012746  006034           MOV     #THR.FMT,-(SP)
14773 052270  012746  000004           MOV     #4,-(SP)
14774 052274  010600                   MOV     SP,R0                   ; SP,*
14775 052276  104414                   TRAP    14
14776 052300  010116                   MOV     R1,(SP)                 ; ARR.SEL,*                             7014
14777 052302  012746  004644           MOV     #FMT.9,-(SP)
14778 052306  012746  000002           MOV     #2,-(SP)
14779 052312  010600                   MOV     SP,R0                   ; SP,*
14780 052314  104414                   TRAP    14
```

```
14782                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
14783                                   ;                                                       14-May-1981 13:05:08    PA:<
14784
14785 052316  012766  000001  000026            MOV     #1,26(SP)               ; *,DODU.FLG                            7015
14786 052324  062706  000016                    ADD     #16,SP                  ;                                       7011
14787 052330  104467                    22$:    TRAP    67                      ;                                       7016
14788 052332  006000                            ROR     R0
14789 052334  103002                            BHIS    23$
14790 052336  000167  177142                    JMP     2$
14791 052342  026627  000010  000001    23$:    CMP     10(SP),#1               ; DODU.FLG,*                            7020
14792 052350  001004                            BNE     24$
14793 052352  016700  141222                    MOV     ML.LUN,R0               ;                                       7023
14794 052356  104451                            TRAP    51
14795 052360  104444                            TRAP    44
14796 052362  066601  000004            24$:    ADD     4(SP),R1                ; *,ARR.SEL                             6954
14797 052366  020166  000006            25$:    CMP     R1,6(SP)                ; ARR.SEL,*
14798 052372  003002                            BGT     26$
14799 052374  000167  177100                    JMP     1$
14800 052400  062706  000016            26$:    ADD     #16,SP                  ;                                       6901
14801 052404  000207                            RTS     PC
14802
14803                                   ; Routine Size:  244 words
14804                                   ; Maximum stack depth per invocation:  20 words
14809
14810
14814
14815                                           .SBTTL  T28
14819 052406                            T28::
14820 052406  004767  177024            1$:     JSR     PC,$T28                 ;                                       7027
14821 052412  104466                            TRAP    66
14822 052414  006000                            ROR     R0
14823 052416  103773                            BLO     1$
14824 052420  000207                            RTS     PC
14825
14826                                   ; Routine Size:  6 words
14827                                   ; Maximum stack depth per invocation:  0 words
14832
14833
14834 ;           7030  !<BLF/PAGE>
```

```
14836 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
14837 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (79)
14838
14839 ;          7031  !
14840 ;          7032  BGNTST;
14841 ;          7033
14842 ;          7034  !++
14843 ;          7035  ! TEST NUMBER:  TST 29
14844 ;          7036  !
14845 ;          7037  ! TEST NAME:    READ WRITE ARRAYS WITH PROM DATA
14846 ;          7038  !
14847 ;          7039  ! TEST DESCRIPTION:
14848 ;          7040  !       COMBINE THE READING OF ARRAY
14849 ;          7041  !       MODULE DATA WITH ARRAY MODULE UV PROM DATA AND
14850 ;          7042  !       FIND A GOOD BLOCK WHERE FURTHER
14851 ;          7043  !       TESTING WILL BE PERFORMED BY:
14852 ;          7044  !
14853 ;          7045  !       1.  STARTING AT BLOCK 0 WRITE THE BLOCK WITH SELECTED DATA PATTERNS
14854 ;          7046  !           AND READ THE BLOCK AVOIDING ANY BAD NIBBLES POINTED TO BY THE
14855 ;          7047  !           PROM DATA.
14856 ;          7048  !
14857 ;          7049  !           SET ERROR FLAG IF ANY BAD NIBBLES ARE ENCOUNTERED IN BLOCK.
14858 ;          7050  !
14859 ;          7051  !       2.  REPEAT WRITING/READING THIS BLOCK UNTIL ALL PATTERNS ARE TESTED
14860 ;          7052  !           OR THE ERROR FLAG IS SET.
14861 ;          7053  !
14862 ;          7054  !       3.  IF ALL PATTERN HAVE BEEN TESTED AND THE ERROR FLAG IS NOT SET
14863 ;          7055  !           THEN SAVE THIS BLOCK ADDRESS AS THE GOOD BLOCK ADRS AND EXIT TEST.
14864 ;          7056  !
14865 ;          7057  !       4.  ELSE IF THE ERROR FLG HAS SET THEN REPEAT TEST AT THE NEXT ROW.
14866 ;          7058  !           REPEAT UNTIL A GOOD BLOCK IS FOUND OR LAST ROW IS REACHED.
14867 ;          7059  !
14868 ;          7060  !       5.  IF NO GOOD BLOCK IS FOUND BY LAST ROW THEN REPORT ERROR AND
14869 ;          7061  !           EXIT TEST.
14870 ;          7062  !
14871 ;          7063  ! IMPLICIT INPUTS:
14872 ;          7064  !       RAS_INC
14873 ;          7065  !       LOADED DURING THE INITIALIZATION CODE AND CONTAINS THE ROW ADDRESS
14874 ;          7066  !       INCREMENT VALUE FOR THIS DRIVE.
14875 ;          7067  !
14876 ;          7068  !
14877 ;          7069  !       PD_TEMP:
14878 ;          7070  !       A BITVECTOR OF 16 BITS WHERE THE READ PROM DATA IS STORED AND
14879 ;          7071  !       ACCESSED FROM.
14880 ;          7072  !
14881 ;          7073  !       IO_BUF:
14882 ;          7074  !       A VECTOR OF 256 WORDS WHERE DATA FOR MBUS READS AND WRITE FUNCTION
14883 ;          7075  !       ARE FOUND.
14884 ;          7076  !
14885 ;          7077  !
14886 ;          7078  !--
14887 ;          7079
14888 ;          7080  local
14889 ;          7081      WRD_CNT,                          !WORD COUNT
14890 ;          7082      NIB_PTR,                          !NIBBLE POINTER
14891 ;ML4                                                   14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
14892 ;                                                      14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (79)
```

```
14893
14894 ;          7083      PASS_CNT,                                   !PASS COUNT
14895 ;          7084      NIB_PAT,                                    !NIBBLE PATTERN
14896 ;          7085      DONE_FLG,                                   !DONE FLAG
14897 ;          7086      ERR_FLG,                                    !ERROR FLAG
14898 ;          7087      SECTOR_NO;                                  !SECTOR NUMBER
14899 ;          7088
14900 ;          7089  BGNSUB;
14901 ;          7090  PASS_CNT = -1;
14902 ;          7091  SECTOR_NO = ZEROES;
14903 ;          7092  DONE_FLG = ZERO;
14904 ;          7093
14905 ;          7094  do                                              !THIS LOOP DETERMINES WHEN TO STOP
14906 ;          7095      begin
14907 ;          7096
14908 ;          7097      do                                          !THIS LOOP RUNS THE PATTERNS
14909 ;          7098          begin
14910 ;          7099          PASS_CNT = .PASS_CNT + 1;               !INCREMENT THE PASS COUNT
14911 ;          7100          CLR_MBUS;
14912 ;          7101          DAT_DM = ONE;                           !SET DATA DIAG MODE
14913 ;          7102          MLDA = .SECTOR_NO;                      !LOAD SECTOR NUMBER IN DSA
14914 ;          7103          MLWC = not 255;                         !LOAD WORD COUNT
14915 ;          7104          MLBA = IO_BUF;                          !LOAD UBUS ADRS
14916 ;          7105
14917 ;          7106          case .PASS_CNT from 0 to 3 of           !SELECT A NIBBLE PATTERN
14918 ;          7107              set
14919 ;          7108
14920 ;          7109              [0] :
14921 ;          7110                  NIB_PAT = %o'000000';           !ZEROES
14922 ;          7111
14923 ;          7112              [1] :
14924 ;          7113                  NIB_PAT = %o'17';               !ONES
14925 ;          7114
14926 ;          7115              [2] :
14927 ;          7116                  NIB_PAT = %o'12';               !ALTERNATING ONE'S, ZEROES
14928 ;          7117
14929 ;          7118              [3] :
14930 ;          7119                  NIB_PAT = %o'15'                !COMPLIMENT ONE'S, ZEROES
14931 ;          7120              tes;
14932 ;          7121
14933 ;          7122          incr LD_CNT from 0 to 9 do              !LOAD NIBBLE PATTERN INTO NIBBLE SAVE
14934 ;          7123              LD_LNG_WRD (.LD_CNT, .NIB_PAT);
14935 ;          7124
14936 ;          7125          WRT_LNG_WRD;                            !LOAD THE DATA DIAG REGISTERS WITH NIBBLE SAVE
1493  ;          7126          MLCS1 = write;                          !DO A WRITE FUNCTION
1493  ;          7127
14939 ;          7128          incr WRT_CNT from 0 to 127 do           !WRITE PATTERN INTO THIS BLOCK
14940 ;          7129              begin
14941 ;          7130              DELAY (ONE_US);
14942 ;          7131              DAT_CLK = ONE;
14943 ;          7132              end;
14944 ;          7133
14945 ;          7134          CLR_MBUS;
14946 ;ML4                                                             14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
14947 ;                                                               14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (79)
14948
14949 ;          7135          DAT_DM = ONE;                           !SET DATA DIAG MODE
```

```
14950 ;          7136          MLDA = .SECTOR_NO;                          !LOAD SECTOR NUMBER
14951 ;          7137          MLWC = not 255;                             !LOAD WORD COUNT
14952 ;          7138          MLBA = IO_BUF;                              !LOAD UBUS ADRS
14953 ;          7139          MLCS1 = read;                               !DO A READ FUNCTION
14954 ;          7140          DELAY (ONE_US);
14955 ;          7141          WRD_CNT = -1;                               !RESET THE WORD COUNT
14956 ;          7142
14957 ;          7143          do
14958 ;          7144              begin                                   !TEST ONE BLOCK FOR NIBBLE PATTERN
14959 ;          7145              WRD_CNT = .WRD_CNT + 1;
14960 ;          7146              PD_TEMP = .MLPD;                        !GET PROM DATA
14961 ;          7147              DAT_CLK = ONE;                          !CLOCK DATA WORD INTO DIAG REG
14962 ;          7148              DELAY (ONE_US);                         !
14963 ;          7149              RD_LNG_WRD;                             !READ DIAG REGISTERS
14964 ;          7150              NIB_PTR = -1;                           !RESET THE NIBBLE POINTER
14965 ;          7151
14966 ;          7152              do
14967 ;          7153                  begin                               !SEARCH FOR GOOD NIBBLES
14968 ;          7154                  NIB_PTR = .NIB_PTR + 1;             !INCREMENT NIBBLE POINTER
14969 ;          7155
14970 ;          7156                  if .PD_TEMP [.NIB_PTR] IS_NOT_SET       !TEST THIS NIB IF PROM FLAG NOT SET
14971 ;          7157                  then
14972 ;          7158                      TST_LNG_WRD (.NIB_PTR, .NIB_PAT, ERR_FLG);
14973 ;          7159
14974 ;          7160                  end
14975 ;          7161              until (.ERR_FLG) or (.NIB_PTR eql 9);
14976 ;          7162
14977 ;          7163              end
14978 ;          7164          until (.ERR_FLG) or (.WRD_CNT eql 127);
14979 ;          7165
14980 ;          7166          end
14981 ;          7167      until (.PASS_CNT eql 3) or (.ERR_FLG IS_SET );      !REPEAT UNTIL ALL PAT TESTED OR ERROR FLG GETS SET
14982 ;          7168
14983 ;          7169      if (.PASS_CNT eql 3) and (.ERR_FLG IS_NOT_SET )     !WAS THIS A GOOD BLOCK?
14984 ;          7170      then
14985 ;          7171          begin                                       !YES
14986 ;          7172          DONE_FLG = ONE;                             !SET DONE FLAG
14987 ;          7173          GOOD_BLK = .SECTOR_NO;                      !GOOD BLOCK GETS THIS SECTOR NO
14988 ;          7174          end
14989 ;          7175      else
14990 ;          7176          begin                                       !NO
14991 ;          7177          SECTOR_NO = .SECTOR_NO + .RAS_INC;          !INCREMENT ROW NO
14992 ;          7178          PASS_CNT = -1;                              !RESET PASS COUNT
14993 ;          7179          end;
14994 ;          7180
14995 ;          7181      end
14996 ;          7182  until (.DONE_FLG IS_SET ) or (.SECTOR_NO eql .LST_ARR + .ARR_INC);
14997 ;          7183
14998 ;          7184                                                      !REPEAT UNTIL GOOD BLK FOUND OR AT LST ROW
14999 ;          7185  ENDSUB;
15000 ;          7186
15001 ;ML4                                                                14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
15002 ;                                                                   14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (79)
15003
15004 ;          7187  if .SECTOR_NO eql .LST_ARR + .ARR_INC              !SEE IF WE'RE AT THE LAST BLOCK
15005 ;          7188  then
15006 ;          7189      begin                                           !A GOOD BLK NOT FOUND BEFORE LAST BLK
```

```
 15007 ;          7190      ERRDF (79, INTER, DUMPER);                    !ERROR AND DROP UNIT
 15008 ;          7191      PRINTB (THR_FMT, WRD_14, PHR_10, FNC_15);
 15009 ;          7192      DODU (.ML_LUN);
 15010 ;          7193      DOCLN;
 15011 ;          7194      end;
 15012 ;          7195
 15013 ;          7196  ENDTST;
 15017
 15018                                          .SBTTL  $T29
 15022 052422  004167  131470          $T29:   JSR     R1,$SAVE5                ;                                  7029
 15023 052426  162706  000014                  SUB     #14,SP
 15024 052432  104402                  1$:     TRAP    2                        ;                                  7087
 15025 052434  012766  177777  000002          MOV     #-1,2(SP)                ; *,PASS.CNT                       7090
 15026 052442  005002                          CLR     R2                       ; SECTOR.NO                        7091
 15027 052444  005066  000004                  CLR     4(SP)                    ; DONE.FLG                         7092
 15028 052450  005266  000002          2$:     INC     2(SP)                    ; PASS.CNT                         7099
 15029 052454  152777  000040  140532          BISB    #40,@ML.REG+40
 15030 052462  016705  141114                  MOV     ML.DUT,R5
 15031 052466  042705  177770                  BIC     #177770,R5
 15032 052472  142777  000007  140514          BICB    #7,@ML.REG+40
 15033 052500  150577  140510                  BISB    R5,@ML.REG+40
 15034 052504  152777  000010  140562          BISB    #10,@ML.REG+120          ;                                  7101
 15035 052512  010277  140466                  MOV     R2,@ML.REG+30            ; SECTOR.NO,*                      7102
 15036 052516  012777  177400  140440          MOV     #-400,@ML.REG+10         ;                                  7103
 15037 052524  012777  011600  140442          MOV     #IO.BUF,@ML.REG+20       ;                                  7104
 15038 052532  016605  000002                  MOV     2(SP),R5                 ; PASS.CNT,*                       7106
 15039 052536  006305                          ASL     R5
 15040 052540  066507  052544                  ADD     3$(R5),PC
 15041 052544  000010                  3$:     .WORD   4$-3$
 15042 052546  000014                          .WORD   5$-3$
 15043 052550  000022                          .WORD   6$-3$
 15044 052552  000030                          .WORD   7$-3$
 15045 052554  005001                  4$:     CLR     R1                       ; NIB.PAT                          7110
 15046 052556  000410                          BR      8$                       ;                                  7106
 15047 052560  012701  000017          5$:     MOV     #17,R1                   ; *,NIB.PAT                        7113
 15048 052564  000405                          BR      8$                       ;                                  7106
 15049 052566  012701  000012          6$:     MOV     #12,R1                   ; *,NIB.PAT                        7116
 15050 052572  000402                          BR      8$                       ;                                  7106
 15051 052574  012701  000015          7$:     MOV     #15,R1                   ; *,NIB.PAT                        7119
 15052 052600  005005                  8$:     CLR     R5                       ; LD.CNT                           7122
 15053 052602  010546                  9$:     MOV     R5,-(SP)                 ; LD.CNT,*                         7123
 15054 052604  010146                          MOV     R1,-(SP)                 ; NIB.PAT,*
 15055 052606  004767  144320                  JSR     PC,LD.LNG.WRD
 15056                                 ;ML4                                                    14-May-1981 13:48:54  TOPS
 15057                                 ;                                                       14-May-1981 13:05:08  PA:<
 15058
 15059 052612  022626                          CMP     (SP)+,(SP)+
 15060 052614  005205                          INC     R5                       ; LD.CNT                           7122
 15061 052616  020527  000011                  CMP     R5,#11                   ; LD.CNT,*
 15062 052622  003767                          BLE     9$
 15063 052624  016777  136322  140512          MOV     D1.TEMP,@ML.REG+170      ;                                  7123
 15064 052632  016777  136316  140514          MOV     D2.TEMP,@ML.REG+200
 15065 052640  016777  136312  140466          MOV     E2.TEMP,@ML.REG+160
 15066 052646  012777  000061  140300          MOV     #61,@ML.REG              ;                                  7126
 15067 052654  005003                          CLR     R3                       ; WRT.CNT                          7128
 15068 052656  012704  000001          10$:    MOV     #1,R4                    ; *,$$TMP2                         7130
 15069 052662  001411                  11$:    BEQ     14$
```

```
15070 052664  016705  127226              MOV     L$DLY,R5                ; *,$$TMP1
15071 052670  001404                      BEQ     13$
15072 052672  005066  000012        12$:  CLR     12(SP)                  ; $$TMP
15073 052676  005305                      DEC     R5                      ; $$TMP1
15074 052700  001374                      BNE     12$
15075 052702  005304              13$:    DEC     R4                      ; $$TMP2
15076 052704  000766                      BR      11$
15077 052706  152777  000020  140360  14$:  BISB    #20,@ML.REG+120         ;                                       7131
15078 052714  005203                      INC     R3                      ; WRT.CNT                               7128
15079 052716  020327  000177              CMP     R3,#177                 ; WRT.CNT,*
15080 052722  003755                      BLE     10$
15081 052724  152777  000040  140262      BISB    #40,@ML.REG+40          ;                                       7132
15082 052732  016705  140644              MOV     ML.DUT,R5
15083 052736  042705  177770              BIC     #177770,R5
15084 052742  142777  000007  140244      BICB    #7,@ML.REG+40
15085 052750  150577  140240              BISB    R5,@ML.REG+40
15086 052754  152777  000010  140312      BISB    #10,@ML.REG+120         ;                                       7135
15087 052762  010277  140216              MOV     R2,@ML.REG+30           ; SECTOR.NO,*                           7136
15088 052766  012777  177400  140170      MOV     #-400,@ML.REG+10        ;                                       7137
15089 052774  012777  011600  140172      MOV     #10.BUF,@ML.REG+20      ;                                       7138
15090 053002  012777  000071  140144      MOV     #71,@ML.REG             ;                                       7139
15091 053010  012704  000001              MOV     #1,R4                   ; *,$$TMP2                              7140
15092 053014  001411              15$:    BEQ     18$
15093 053016  016705  127074              MOV     L$DLY,R5                ; *,$$TMP1
15094 053022  001404                      BEQ     17$
15095 053024  005066  000012        16$:  CLR     12(SP)                  ; $$TMP
15096 053030  005305                      DEC     R5                      ; $$TMP1
15097 053032  001374                      BNE     16$
15098 053034  005304              17$:    DEC     R4                      ; $$TMP2
15099 053036  000766                      BR      15$
15100 053040  012766  177777  000006  18$:  MOV     #-1,6(SP)               ; *,WRD.CNT                             7141
15101 053046  005266  000006        19$:  INC     6(SP)                   ; WRD.CNT                               7145
15102 053052  017767  140326  140040      MOV     @ML.REG+230,PD.TEMP     ;                                       7146
15103 053060  152777  000020  140206      BISB    #20,@ML.REG+120         ;                                       7147
15104 053066  012704  000001              MOV     #1,R4                   ; *,$$TMP2                              7148
15105 053072  001411              20$:    BEQ     23$
15106 053074  016705  127016              MOV     L$DLY,R5                ; *,$$TMP1
15107 053100  001404                      BEQ     22$
15108 053102  005066  000012        21$:  CLR     12(SP)                  ; $$TMP
15109 053106  005305                      DEC     R5                      ; $$TMP1
15110 053110  001374                      BNE     21$
15111                             ;ML4                                                    14-May-1981 13:48:54    TOPS
15112                             ;                                                       14-May-1981 13:05:08    PA:<
15113
15114 053112  005304              22$:    DEC     R4                      ; $$TMP2
15115 053114  000766                      BR      20$
15116 053116  017767  140222  136026  23$:  MOV     @ML.REG+170,D1.TEMP
15117 053124  017767  140224  136022      MOV     @ML.REG+200,D2.TEMP
15118 053132  017767  140176  136016      MOV     @ML.REG+160,E2.TEMP
15119 053140  012716  177777              MOV     #-1,(SP)                ; *,NIB.PTR                             7150
15120 053144  005216              24$:    INC     (SP)                    ; NIB.PTR                               7154
15121 053146  011605                      MOV     (SP),R5                 ; NIB.PTR,*                             7156
15122 053150  006205                      ASR     R5
15123 053152  006205                      ASR     R5
15124 053154  006205                      ASR     R5
15125 053156  062705  013120              ADD     #PD.TEMP,R5
15126 053162  010546                      MOV     R5,-(SP)
```

```
15127 053164  016646  000002                MOV     2(SP),-(SP)             ; NIB.PTR,*
15128 053170  042716  177770                BIC     #177770,(SP)
15129 053174  012746  000001                MOV     #1,-(SP)
15130 053200  005046                        CLR     -(SP)
15131 053202  004767  127732                JSR     PC,BL$GT2
15132 053206  062706  000010                ADD     #10,SP
15133 053212  005700                        TST     R0
15134 053214  001011                        BNE     25$
15135 053216  011646                        MOV     (SP),-(SP)              ; NIB.PTR,*                                     7158
15136 053220  010146                        MOV     R1,-(SP)                ; NIB.PAT,*
15137 053222  012746  000016                MOV     #16,-(SP)
15138 053226  060616                        ADD     SP,(SP)                 ; ERR.FLG,*
15139 053230  004767  142720                JSR     PC,TST.LNG.WRD
15140 053234  062706  000006                ADD     #6,SP
15141 053240  016605  000010          25$:  MOV     10(SP),R5               ; ERR.FLG,*                                     7161
15142 053244  032705  000001                BIT     #1,R5
15143 053250  001012                        BNE     26$
15144 053252  021627  000011                CMP     (SP),#11                ; NIB.PTR,*
15145 053256  001332                        BNE     24$
15146 053260  032705  000001                BIT     #1,R5                   ;                                               7164
15147 053264  001004                        BNE     26$
15148 053266  026627  000006  000177        CMP     6(SP),#177              ; WRD.CNT,*
15149 053274  001264                        BNE     19$
15150 053276  005004                  26$:  CLR     R4                      ;                                               7167
15151 053300  026627  000002  000003        CMP     2(SP),#3                ; PASS.CNT,*
15152 053306  001002                        BNE     27$
15153 053310  005204                        INC     R4
15154 053312  000403                        BR      28$
15155 053314  020527  000001          27$:  CMP     R5,#1
15156 053320  001030                        BNE     31$
15157 053322  006004                  28$:  ROR     R4                      ;                                               7169
15158 053324  103010                        BCC     29$
15159 053326  005705                        TST     R5
15160 053330  001006                        BNE     29$
15161 053332  012766  000001  000004        MOV     #1,4(SP)                ; *,DONE.FLG                                    7172
15162 053340  010267  136220                MOV     R2,GOOD.BLK             ; SECTOR.NO,*                                   7173
15163 053344  000405                        BR      30$                     ;                                               7169
15164 053346  066702  137552          29$:  ADD     RAS.INC,R2              ; *,SECTOR.NO                                   7177
15165 053352  012766  177777  000002        MOV     #-1,2(SP)               ; *,PASS.CNT                                    7178
15166                               ;ML4                                                  14-May-1981 13:48:54    TOPS
15167                               ;                                                     14-May-1981 13:05:08    PA:<
15168
15169 053360  026627  000004  000001  30$:  CMP     4(SP),#1                ; DONE.FLG,*                                    7182
15170 053366  001410                        BEQ     32$
15171 053370  016705  136202                MOV     LST.ARR,R5
15172 053374  066705  136162                ADD     ARR.INC,R5
15173 053400  020205                        CMP     R2,R5                   ; SECTOR.NO,*
15174 053402  001402                  31$:  BEQ     32$
15175 053404  000167  177040                JMP     2$
15176 053410  104467                  32$:  TRAP    67
15177 053412  006000                        ROR     R0
15178 053414  103002                        BHIS    33$
15179 053416  000167  177010                JMP     1$
15180 053422  016705  136150          33$:  MOV     LST.ARR,R5              ;                                               7187
15181 053426  066705  136130                ADD     ARR.INC,R5
15182 053432  020205                        CMP     R2,R5                   ; SECTOR.NO,*
15183 053434  001026                        BNE     34$
```

```
15184 053436  104455                    TRAP    55                              ;                                       7190
15185 053440  000117                    .WORD   117
15186 053442  010672                    .WORD   INTER
15187 053444  024060                    .WORD   DUMPER
15188 053446  012746  010136            MOV     #FNC.15,-(SP)                   ;                                       7191
15189 053452  012746  007564            MOV     #PHR.10,-(SP)
15190 053456  012746  006442            MOV     #WRD.14,-(SP)
15191 053462  012746  006034            MOV     #THR.FMT,-(SP)
15192 053466  012746  000004            MOV     #4,-(SP)
15193 053472  010600                    MOV     SP,R0                           ; SP,*
15194 053474  104414                    TRAP    14
15195 053476  016700  140076            MOV     ML.LUN,R0                       ;                                       7192
15196 053502  104451                    TRAP    51
15197 053504  104444                    TRAP    44
15198 053506  062706  000012            ADD     #12,SP                          ;                                       7189
15199 053512  062706  000014    34$:    ADD     #14,SP                          ;                                       7029
15200 053516  000207                    RTS     PC
15201
15202                                 ; Routine Size:  287 words
15203                                 ; Maximum stack depth per invocation:  17 words
15208
15209
15213
15214                                           .SBTTL  T29
15218 053520                          T29::
15219 053520  004767  176676    1$:     JSR     PC,$T29                         ;                                       7194
15220 053524  104466                    TRAP    66
15221                                 ;ML4                                                    14-May-1981 13:48:54    TOPS
15222                                 ;                                                       14-May-1981 13:05:08    PA:<
15223
15224 053526  006000                            ROR     R0
15225 053530  103773                            BLO     1$
15226 053532  000207                            RTS     PC
15227
15228                                 ; Routine Size:  6 words
15229                                 ; Maximum stack depth per invocation:  0 words
15234
15235
15236 ;          7197  !<BLF/PAGE>
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (80)

```
15238 ;ML4
15239 ;
15240
15241 ;          7198  !
15242 ;          7199  !
15243 ;          7200  BGNTST;
15244 ;          7201
15245 ;          7202  !++
15246 ;          7203  ! TEST NUMBER:  TST 30
15247 ;          7204  !
15248 ;          7205  ! TEST NAME:    REFRESH TIMING TEST
15249 ;          7206  !
15250 ;          7207  ! TEST DESCRIPTION:
15251 ;          7208  !
15252 ;          7209  !       TEST THE MEMORY ARRAY MODULES
15253 ;          7210  !       REFRESH TIMING AND CONTROL LOGIC
15254 ;          7211  !       REG BY:
15255 ;          7212  !
15256 ;          7213  !       1.  WRITING ALL ONES INTO THE
15257 ;          7214  !           GOOD BLOCK.
15258 ;          7215  !
15259 ;          7216  !       2.  DELAY FOR 2 MS
15260 ;          7217  !
15261 ;          7218  !       3.  READ THE GOOD BLOCK FOR ONES.
15262 ;          7219  !           KEEP COUNT OF BAD NIBBLES
15263 ;          7220  !           ENCOUNTERED
15264 ;          7221  !
15265 ;          7222  !       4.  ALLOW A BAD NIBBLE THRESHOLD
15266 ;          7223  !           OF 20 BAD NIBBLES OUT OF THE
15267 ;          7224  !           1280 NIBBLES TESTED.
15268 ;          7225  !
15269 ;          7226  !       5.  IF THRESHOLD IS EXCEEDED THEN
15270 ;          7227  !           REPORT ERROR AND DROP THE UNIT.
15271 ;          7228  !
15272 ;          7229  ! IMPLICIT INPUTS:
15273 ;          7230  !
15274 ;          7231  !       PD_TEMP:
15275 ;          7232  !       A BIT VECTOR OF 16 BITS WHERE
15276 ;          7233  !       THE READ PROM DATA IS STORED
15277 ;          7234  !       AND ACCESSED FROM.
15278 ;          7235  !
15279 ;          7236  !--
15280 ;          7237
15281 ;          7238  local
15282 ;          7239      TST_PAT,                          !TEST PATTERN
15283 ;          7240      ERR_CNT,                          !ERROR COUNT
15284 ;          7241      ERR_FLG;                          !ERROR FLAG
15285 ;          7242
15286 ;          7243  BGNSUB;
15287 ;          7244  CLR_MBUS;
15288 ;          7245  ERR_CNT = ZERO;
15289 ;          7246  TST_PAT = ONES;
15290 ;          7247  MLD1 = .TST_PAT;                      !LOAD DATA DIAG REGS WITH TST PAT
15291 ;          7248  MLD2 = .TST_PAT;
15292 ;          7249  MLE2 = .TST_PAT;
```

```
15294 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
15295 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (80)
15296
15297 ;          7250  DAT_DM_XFER ();                        !SET UP A DATA DIAG MODE XFERR
15298 ;          7251  MLCS1 = write;                         !DO A WRITE FUNCTION
15299 ;          7252
15300 ;          7253  incr WRD_CNT from 0 to 127 do          !WRITE BLOCK WITH TST PAT
15301 ;          7254      begin
15302 ;          7255      DELAY (ONE_US);
15303 ;          7256      DAT_CLK = ONE;
15304 ;          7257      end;
15305 ;          7258
15306 ;          7259  CLR_MBUS;
15307 ;          7260  DAT_DM_XFER ();                        !SET UP A DATA DIAG MODE XFERR
15308 ;          7261  MLCS1 = read;                          !DO A READ FUNCTION
15309 ;          7262  DELAY (TWO_TH_US);                     !DELAY FOR 2 MS
15310 ;          7263
15311 ;          7264  incr WRD_CNT from 0 to 127 do          !READ THE BLOCK
15312 ;          7265      begin
15313 ;          7266      PD_TEMP = .MLPD;                   !GET THE PROM DATA
15314 ;          7267      DAT_CLK = ONE;                     !CLOCK DATA WORD INTO DIAG REG
15315 ;          7268      DELAY (ONE_US);
15316 ;          7269      RD_LNG_WRD;                        !READ THE DIAG REG
15317 ;          7270
15318 ;          7271      incr NIB_PTR from 0 to 9 do        !LOOK AT 10 NIBBLES
15319 ;          7272          begin
15320 ;          7273
15321 ;          7274          if .PD_TEMP [.NIB_PTR] IS_NOT_SET then TST_LNG_WRD (.NIB_PTR, .TST_PAT, ERR_FLG);
15322 ;          7275
15323 ;          7276                                         !FIND GOOD NIBBLES AND COMPARE THEM
15324 ;          7277
15325 ;          7278          if .ERR_FLG IS_SET then ERR_CNT = .ERR_CNT + 1; !INCREMENT ERROR COUNT IF ERR_FLG IS SET
15326 ;          7279
15327 ;          7280          end;
15328 ;          7281
15329 ;          7282      end;
15330 ;          7283
15331 ;          7284  ENDSUB;
15332 ;          7285
15333 ;          7286  if .ERR_CNT gtr 20                     !ALLOW 20 NIBBLES TO FAIL
15334 ;          7287  then
15335 ;          7288      begin                              !ERROR IF GTR 20
15336 ;          7289      ERRDF (80, ASYNC, DUMPER);
15337 ;          7290      PRINTB (FIV_FMT, WRD_22, PHR_4, WRD_12, FNC_16, WRD_48);
15338 ;          7291      DODU (.ML_LUN);
15339 ;          7292      DOCLN;
15340 ;          7293      end;
15341 ;          7294
15342 ;          7295  ENDTST;
15346
15347                                  .SBTTL  $T30
```

```
15355 053534  004167  130356          $T30:   JSR     R1,$SAVE5                ;                                  7196
15356 053540  024646                          CMP     -(SP),-(SP)
15357 053542  104402                  1$:     TRAP    2                        ;                                  7241
15358 053544  152777  000040  137442          BISB    #40,@ML.REG+40           ;                                  7243
15359 053552  016703  140024                  MOV     ML.DUT,R3
15360 053556  042703  177770                  BIC     #177770,R3
15361 053562  142777  000007  137424          BICB    #7,@ML.REG+40
15362 053570  150377  137420                  BISB    R3,@ML.REG+40
15363 053574  005005                          CLR     R5                       ; ERR.CNT                          7245
15364 053576  012704  177777                  MOV     #-1,R4                   ; *,TST.PAT                        7246
15365 053602  010477  137536                  MOV     R4,@ML.REG+170           ; TST.PAT,*                        7247
15366 053606  010477  137542                  MOV     R4,@ML.REG+200           ; TST.PAT,*                        7248
15367 053612  010477  137516                  MOV     R4,@ML.REG+160           ; TST.PAT,*                        7249
15368 053616  004767  140340                  JSR     PC,DAT.DM.XFER           ;                                  7250
15369 053622  012777  000061  137324          MOV     #61,@ML.REG              ;                                  7251
15370 053630  005001                          CLR     R1                       ; WRD.CNT                          7253
15371 053632  012702  000001          2$:     MOV     #1,R2                    ; *,$$TMP2                         7255
15372 053636  001411                  3$:     BEQ     6$
15373 053640  016703  126252                  MOV     L$DLY,R3                 ; *,$$TMP1
15374 053644  001404                          BEQ     5$
15375 053646  005066  000002          4$:     CLR     2(SP)                    ; $$TMP
15376 053652  005303                          DEC     R3                       ; $$TMP1
15377 053654  001374                          BNE     4$
15378 053656  005302                  5$:     DEC     R2                       ; $$TMP2
15379 053660  000766                          BR      3$
15380 053662  152777  000020  137404  6$:     BISB    #20,@ML.REG+120          ;                                  7256
15381 053670  005201                          INC     R1                       ; WRD.CNT                          7253
15382 053672  020127  000177                  CMP     R1,#177                  ; WRD.CNT,*
15383 053676  003755                          BLE     2$
15384 053700  152777  000040  137306          BISB    #40,@ML.REG+40           ;                                  7257
15385 053706  016703  137670                  MOV     ML.DUT,R3
15386 053712  042703  177770                  BIC     #177770,R3
15387 053716  142777  000007  137270          BICB    #7,@ML.REG+40
15388 053724  150377  137264                  BISB    R3,@ML.REG+40
15389 053730  004767  140226                  JSR     PC,DAT.DM.XFER           ;                                  7260
15390 053734  012777  000071  137212          MOV     #71,@ML.REG              ;                                  7261
15391 053742  012702  003720                  MOV     #3720,R2                 ; *,$$TMP2                         7262
15392 053746  001411                  7$:     BEQ     10$
15393 053750  016703  126142                  MOV     L$DLY,R3                 ; *,$$TMP1
15394 053754  001404                          BEQ     9$
15395 053756  005066  000002          8$:     CLR     2(SP)                    ; $$TMP
15396 053762  005303                          DEC     R3                       ; $$TMP1
15397 053764  001374                          BNE     8$
15398 053766  005302                  9$:     DEC     R2                       ; $$TMP2
15399 053770  000766                          BR      7$
15400 053772  005001                  10$:    CLR     R1                       ; WRD.CNT                          7264
15401 053774  017767  137404  137116  11$:    MOV     @ML.REG+230,PD.TEMP      ;                                  7266
15402 054002  152777  000020  137264          BISB    #20,@ML.REG+120          ;                                  7267
15403 054010  012702  000001                  MOV     #1,R2                    ; *,$$TMP2                         7268
15404 054014  001411                  12$:    BEQ     15$
```

```
15406                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
15407                                   ;                                                       14-May-1981 13:05:08    PA:<
15408
15409 054016  016703  126074                    MOV     L$DLY,R3                ; *,$$TMP1
15410 054022  001404                            BEQ     14$
15411 054024  005066  000002              13$:  CLR     2(SP)                   ; $$TMP
15412 054030  005303                            DEC     R3                      ; $$TMP1
15413 054032  001374                            BNE     13$
15414 054034  005302                      14$:  DEC     R2                      ; $$TMP2
15415 054036  000766                            BR      12$
15416 054040  017767  137300  135104      15$:  MOV     @ML.REG+170,D1.TEMP
15417 054046  017767  137302  135100            MOV     @ML.REG+200,D2.TEMP
15418 054054  017767  137254  135074            MOV     @ML.REG+160,E2.TEMP
15419 054062  005002                            CLR     R2                      ; NIB.PTR                               7271
15420 054064  010203                      16$:  MOV     R2,R3                   ; NIB.PTR,*                             7274
15421 054066  006203                            ASR     R3
15422 054070  006203                            ASR     R3
15423 054072  006203                            ASR     R3
15424 054074  062703  013120                    ADD     #PD.TEMP,R3
15425 054100  010346                            MOV     R3,-(SP)
15426 054102  010246                            MOV     R2,-(SP)                ; NIB.PTR,*
15427 054104  042716  177770                    BIC     #177770,(SP)
15428 054110  012746  000001                    MOV     #1,-(SP)
15429 054114  005046                            CLR     -(SP)
15430 054116  004767  127016                    JSR     PC,BL$GT2
15431 054122  062706  000010                    ADD     #10,SP
15432 054126  005700                            TST     R0
15433 054130  001011                            BNE     17$
15434 054132  010246                            MOV     R2,-(SP)                ; NIB.PTR,*
15435 054134  010446                            MOV     R4,-(SP)                ; TST.PAT,*
15436 054136  012746  000006                    MOV     #6,-(SP)
15437 054142  060616                            ADD     SP,(SP)                 ; ERR.FLG,*
15438 054144  004767  142004                    JSR     PC,TST.LNG.WRD
15439 054150  062706  000006                    ADD     #6,SP
15440 054154  021627  000001        17$:  CMP     (SP),#1                 ; ERR.FLG,*                             7278
15441 054160  001001                            BNE     18$
15442 054162  005205                            INC     R5                      ; ERR.CNT
15443 054164  005202                      18$:  INC     R2                      ; NIB.PTR                               7271
15444 054166  020227  000011                    CMP     R2,#11                  ; NIB.PTR,*
15445 054172  003734                            BLE     16$
15446 054174  005201                            INC     R1                      ; WRD.CNT                               7264
15447 054176  020127  000177                    CMP     R1,#177                 ; WRD.CNT,*
15448 054202  003674                            BLE     11$
15449 054204  104467                            TRAP    67                      ;                                       7282
15450 054206  006000                            ROR     R0
15451 054210  103002                            BHIS    19$
15452 054212  000167  177324                    JMP     1$
15453 054216  020527  00002?        19$:  CMP     R5,#24                  ; ERR.CNT,*                             7286
15454 054222  003432                            BLE     20$
15455 054224  104455                            TRAP    55                      ;                                       7289
15456 054226  000120                            .WORD   120
15457 054230  010464                            .WORD   ASYNC
15458 054232  024060                            .WORD   DUMPER
15459 054234  012746  007026                    MOV     #WRD.48,-(SP)           ;                                       7290
15460 054240  012746  010150                    MOV     #FNC.16,-(SP)
```

```
15462                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
15463                                   ;                                                       14-May-1981 13:05:08    PA:<
15464
15465 054244  012746  006426                    MOV     #WRD.12,-(SP)
15466 054250  012746  007454                    MOV     #PHR.4,-(SP)
15467 054254  012746  006530                    MOV     #WRD.22,-(SP)
15468 054260  012746  006062                    MOV     #FIV.FMT,-(SP)
15469 054264  012746  000006                    MOV     #6,-(SP)
15470 054270  010600                            MOV     SP,R0                           ; SP,*
15471 054272  104414                            TRAP    14
15472 054274  016700  137300                    MOV     ML.LUN,R0                       ;                       7291
15473 054300  104451                            TRAP    51
15474 054302  104444                            TRAP    44
15475 054304  062706  000016                    ADD     #16,SP                          ;                       7288
15476 054310  022626                    20$:    CMP     (SP)+,(SP)+                     ;                       7196
15477 054312  000207                            RTS     PC
15478
15479                                   ; Routine Size:  184 words
15480                                   ; Maximum stack depth per invocation:  15 words
15485
15486
15490
15491                                           .SBTTL  T30
15495 054314                            T30::
15496 054314  004767  177214            1$:     JSR     PC,$T30                         ;                       7293
15497 054320  104466                            TRAP    66
15498 054322  006000                            ROR     R0
15499 054324  103773                            BLO     1$
15500 054326  000207                            RTS     PC
15501
15502                                   ; Routine Size:  6 words
15503                                   ; Maximum stack depth per invocation:  0 words
15508
15509
15510 ;           7296  !<BLF/PAGE>
```

```
15512 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
15513 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (81)
15514
15515 ;          7297  !
15516 ;          7298  BGNTST;
15517 ;          7299
15518 ;          7300  !++
15519 ;          7301  ! TEST NUMBER:  TST 31
15520 ;          7302  !
15521 ;          7303  ! TEST NAME:    ADDRESS COUNTER TEST
15522 ;          7304  !
15523 ;          7305  ! TEST DESCRIPTION:
15524 ;          7306  !       TEST THE ABILITY OF THE ADDRESS
15525 ;          7307  !       COUNTER TO SUCCESSFULLY COUNT
15526 ;          7308  !       FROM BLOCK ZERO THROUGH THE
15527 ;          7309  !       DEVICES LAST BLOCK BY:
15528 ;          7310  !
15529 ;          7311  !       1.  WRITING THE LAST BLOCK WITH ONES PATTERN.
15530 ;          7312  !
15531 ;          7313  !       2.  STARTING AT BLOCK ZERO WRITE ZEROES INTO ALL BLOCK UP TO THE
15532 ;          7314  !           LAST BLOCK ADRS MINUS ONE.
15533 ;          7315  !
15534 ;          7316  !           READ THE LAST BLOCK FOR ONES AND ERROR IF ZEROES.
15535 ;          7317  !
15536 ;          7318  !       3.  STARTING AT BLOCK ZERO WRITE ZEROES INTO ALL BLOCK THROUGH THE LAST
15537 ;          7319  !           BLOCK.
15538 ;          7320  !
15539 ;          7321  !           READ THE LAST BLOCK FOR ZEROES AND ERROR IF STILL ONES.
15540 ;          7322  !
15541 ;          7323  ! IMPLICIT INPUTS:
15542 ;          7324  !       PD_TEMP:
15543 ;          7325  !       A BITVECTOR OF 16 BITS WHERE THE READ PROM DATA IS STORED AND ACCESSED FROM.
15544 ;          7326  !--
15545 ;          7327
15546 ;          7328  local
15547 ;          7329      DODU_FLG,                          !DROP UNIT FLAG
15548 ;          7330      ERR_FLG,                           !ERROR FLG
15549 ;          7331      END_CNT,                           !ENDING SECTOR NUMBER
15550 ;          7332      ERR_CNT,                           !COUNTS # OF BAD NIBBLES FOUND IN THE LAST BLOCK
15551 ;          7333      BG_PAT;                            !BACKGROUND PATTERN
15552 ;          7334
15553 ;          7335  BGNSUB;
15554 ;          7336  CLR_MBUS;
15555 ;          7337  DODU_FLG = ZERO;
15556 ;          7338  BG_PAT = ONES;                         !BACKGROUND PAT OF ONES
15557 ;          7339  MLD1 = .BG_PAT;                        !LOAD DATA DIAGS WITH BG PAT
15558 ;          7340  MLD2 = .BG_PAT;
15559 ;          7341  MLE2 = .BG_PAT;
15560 ;          7342  DAT_DM = ONE;                          !SET DATA DIAG MODE
15561 ;          7343  LAST_BLK_XFER ();                      !SET UP A LAST BLOCK XFERR
15562 ;          7344  MLCST = write;                         !DO A WRITE FUNCTION
15563 ;          7345
15564 ;          7346  incr WRD_CNT from 0 to 127 do          !WRITE THE LAST BLOCK WITH THE BG PATTERN
15565 ;          7347      begin
15566 ;          7348      DELAY (ONE_US);
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (81)

;          7349      DAT_CLK = ONE;
;          7350      end;
;          7351
;          7352  CLR_MBUS;
;          7353  DAT_DM = ONE;                          !SET DATA DIAG MODE
;          7354  LAST_BLK_XFER ();                      !SET UP A LAST BLOCK XFERR
;          7355  MLCST = read;                          !DO A READ FUNCTION
;          7356  DELAY (ONE_US);
;          7357  ERR_CNT = ZEROES;                      !CLEAR THE ERROR COUNT
;          7358
;          7359  incr WD_CNT from 0 to 127 do          !READ THE LAST BLOCK FOR BG PATTERN
;          7360      begin
;          7361      PD_TEMP = .MLPD;                   !GET PROM DATA
;          7362      DAT_CLK = ONE;                     !CLOCK OUT THE DATA WORD
;          7363      DELAY (ONE_US);
;          7364      RD_LNG_WRD;                        !READ THE DATA WORD
;          7365
;          7366      incr NIB_PTR from 0 to 9 do        !LOOK AT 10 NIBBLES
;          7367          begin
;          7368
;          7369          if .PD_TEMP [.NIB_PTR] IS_NOT_SET then TST_LNG_WRD (.NIB_PTR, .BG_PAT, ERR_FLG);
;          7370
;          7371                                         !FIND GOOD NIBBLES AND COMPARE THEM
;          7372
;          7373          IF .ERR_FLG IS_SET
;          7374              THEN ERR_CNT = .ERR_CNT + 1;          !COUNT THE # OF BAD NIBBLES DETECTED
;          7375
;          7376          end;
;          7377
;          7378      end;
;          7379
;          7380          if .ERR_CNT GTR 10             !IS ERROR COUNT > THRESHOLD
;          7381          then
;          7382              begin
;          7383              ERRDF (81, INTER, DUMPER);
;          7384              PRINTB (THR_FMT, PHR_4, WRD_13, FNC_22);
;          7385              PRINTB (TWO_FMT, FNC_13, WRD_56);
;          7386              EXIT_TST;
;          7387              end;
;          7388
;          7389  END_CNT = .LST_BLK - 1;                !END AT LAST BLOCK -1
;          7390  MLDT = ZEROES;                         !LOAD DATA DIAG REG WITH COMP BG PAT
;          7391  MLD2 = ZEROES;
;          7392  MLE2 = ZEROES;
;          7393
;          7394  incr TWICE from 0 to 1 do              !REPEAT LOOP TWICE
;          7395      begin
;          7396      CLR_MBUS;
;          7397      ERR_CNT = ZEROES;              !CLEAR THE ERROR COUNT
;          7398      DAT_DM = ONE;                      !SET DATA DIAG MODE
;          7399      FIRST_BLK_XFER ();                 !SET UP A FIRST BLOCK XFERR
;          7400      MLCS1 = write;                     !DO A WRITE FUNCTION
```

```
15624 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
15625 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (81)
15626
15627 ;          7401
15628 ;          7402      incr BLK_CNT from 0 to .END_CNT do          !CLOCK THE ADDRESS COUNTER UP TO END_CNT
15629 ;          7403
15630 ;          7404          incr ADRS_CNT from 0 to 127 do
15631 ;          7405              begin
15632 ;          7406              DAT_CLK = ONE;
15633 ;          7407              end;
15634 ;          7408
15635 ;          7409      CLR_MBUS;
15636 ;          7410      DAT_DM = ONE;                               !SET DATA DIAG MODE
15637 ;          7411      LAST_BLK_XFER ();                           !SET UP A LAST BLOCK XFERR
15638 ;          7412      MLCST = read;                               !DO A READ FUNCTION
15639 ;          7413      DELAY (ONE_US);
15640 ;          7414
15641 ;          7415      incr WD_CNT from 0 to 127 do                !READ THE LAST BLOCK FOR BG PATTERN
15642 ;          7416          begin
15643 ;          7417          PD_TEMP = .MLPD;                        !GET THE PROM DATA
15644 ;          7418          DAT_CLK = ONE;                          !CLOCK OUT DATA WORD
15645 ;          7419          DELAY (ONE_US);
15646 ;          7420          RD_LNG_WRD;                             !READ DATA WORD
15647 ;          7421
15648 ;          7422          incr NIB_PTR from 0 to 9 do             !LOOK AT 10 NIBBLES
15649 ;          7423              begin
15650 ;          7424
15651 ;          7425              if .PD_TEMP [.NIB_PTR] IS_NOT_SET then TST_LNG_WRD (.NIB_PTR, .BG_PAT, ERR_FLG);
15652 ;          7426
15653 ;          7427                                                  !FIND GOOD NIBBLES AND COMPARE THEM
15654 ;          7428
15655 ;          7429              IF .ERR_FLG IS_SET                          !COUNT # OF BAD NIBBLES DETECTED
15656 ;          7430                  THEN ERR_CNT = .ERR_CNT + 1;
15657 ;          7431              end;
15658 ;          7432
15659 ;          7433          end;
15660 ;          7434
15661 ;          7435              IF .ERR_CNT GTR 10                      !IS ERROR COUNT > TRESHOLD
15662 ;          7436              then
15663 ;          7437                  begin
15664 ;          7438                  ERRDF (82, ASYNC, DUMPER);
15665 ;          7439                  PRINTB (THR_FMT, WRD_50, WRD_51, WRD_10);
15666 ;          7440                  DODU_FLG = ONE;
15667 ;          7441                  end;
15668 ;          7442
15669 ;          7443
15670 ;          7444      END_CNT = .END_CNT + 1;                     !NOW END AT THE LAST BLOCK
15671 ;          7445      BG_PAT = not .BG_PAT;                       !COMPLIMENT THE BG PATTERN AND REPEAT
15672 ;          7446      end;
15673 ;          7447
15674 ;          7448  ENDSUB;
15675 ;          7449
15676 ;          7450  if .DODU_FLG IS_SET                            !DROP THIS UNIT IF DODU_FLG IS_SET
15677 ;          7451  then
15678 ;          7452      begin
```

```
15680 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
15681 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (81)
15682
15683 ;          7453       DODU (.ML_LUN);
15684 ;          7454       DOCLN;
15685 ;          7455       end;
15686 ;          7456
15687 ;          7457   ENDTST;
15691
15692                                        .SBTTL  $T31
15696 054330  004167  127562          $T31: JSR     R1,$SAVE5                    ;                                  7295
15697 054334  162706  000012                SUB     #12,SP
15698 054340  104402                  1$:   TRAP    2                            ;                                  7333
15699 054342  152777  000040  136644        BISB    #40,@ML.REG+40               ;                                  7335
15700 054350  016705  137226                MOV     ML.DUT,R5
15701 054354  042705  177770                BIC     #177770,R5
15702 054360  142777  000007  136626        BICB    #7,@ML.REG+40
15703 054366  150577  136622                BISB    R5,@ML.REG+40
15704 054372  005066  000004                CLR     4(SP)                        ; DODU.FLG                         7337
15705 054376  012702  177777                MOV     #-1,R2                       ; *,BG.PAT                         7338
15706 054402  010277  136736                MOV     R2,@ML.REG+170               ; BG.PAT,*                         7339
15707 054406  010277  136742                MOV     R2,@ML.REG+200               ; BG.PAT,*                         7340
15708 054412  010277  136716                MOV     R2,@ML.REG+160               ; BG.PAT,*                         7341
15709 054416  152777  000010  136650        BISB    #10,@ML.REG+120              ;                                  7342
15710 054424  004767  141472                JSR     PC,LAST.BLK.XFER             ;                                  7343
15711 054430  012777  000061  136516        MOV     #61,@ML.REG                  ;                                  7344
15712 054436  005003                        CLR     R3                           ; WRD.CNT                          7346
15713 054440  012704  000001          2$:   MOV     #1,R4                        ; *,$$TMP2                         7348
15714 054444  001411                  3$:   BEQ     6$
15715 054446  016705  125444                MOV     L$DLY,R5                     ; *,$$TMP1
15716 054452  001404                        BEQ     5$
15717 054454  005066  000010          4$:   CLR     10(SP)                       ; $$TMP
15718 054460  005305                        DEC     R5                           ; $$TMP1
15719 054462  001374                        BNE     4$
15720 054464  005304                  5$:   DEC     R4                           ; $$TMP2
15721 054466  000766                        BR      3$
15722 054470  152777  000020  136576  6$:   BISB    #20,@ML.REG+120              ;                                  7349
15723 054476  005203                        INC     R3                           ; WRD.CNT                          7346
15724 054500  020327  000177                CMP     R3,#177                      ; WRD.CNT,*
15725 054504  003755                        BLE     2$
15726 054506  152777  000040  136500        BISB    #40,@ML.REG+40               ;                                  7350
15727 054514  016705  137062                MOV     ML.DUT,R5
15728 054520  042705  177770                BIC     #177770,R5
15729 054524  142777  000007  136462        BICB    #7,@ML.REG+40
15730 054532  150577  136456                BISB    R5,@ML.REG+40
15731 054536  152777  000010  136530        BISB    #10,@ML.REG+120              ;                                  7353
15732 054544  004767  141352                JSR     PC,LAST.BLK.XFER             ;                                  7354
15733 054550  012777  000071  136376        MOV     #71,@ML.REG                  ;                                  7355
15734 054556  012704  000001                MOV     #1,R4                        ; *,$$TMP2                         7356
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                        7$:     BEQ     10$
054562  001411                          MOV     L$DLY,R5                ; *,$$TMP1
054564  016705  125326                  BEQ     9$
054570  001404                  8$:     CLR     10(SP)                  ; $$TMP
054572  005066  000010                  DEC     R5                      ; $$TMP1
054576  005305                          BNE     8$
054600  001374                  9$:     DEC     R4                      ; $$TMP2
054602  005304                          BR      7$
054604  000766                  10$:    CLR     (SP)                    ; ERR.CNT                               7357
054606  005016                          CLR     R3                      ; WD.CNT                                7359
054610  005003          136566  136300  11$:    MOV     @ML.REG+230,PD.TEMP     ;                               7361
054612  017767  136566  136300          BISB    #20,@ML.REG+120         ;                               7362
054620  152777  000020  136446          MOV     #1,R4                   ; *,$$TMP2                              7363
054626  012704  000001          12$:    BEQ     15$
054632  001411                          MOV     L$DLY,R5                ; *,$$TMP1
054634  016705  125256                  BEQ     14$
054640  001404                  13$:    CLR     10(SP)                  ; $$TMP
054642  005066  000010                  DEC     R5                      ; $$TMP1
054646  005305                          BNE     13$
054650  001374                  14$:    DEC     R4                      ; $$TMP2
054652  005304                          BR      12$
054654  000766          134266  15$:    MOV     @ML.REG+170,D1.TEMP
054656  017767  136462          134262  MOV     @ML.REG+200,D2.TEMP
054664  017767  136464          134256  MOV     @ML.REG+160,E2.TEMP
054672  017767  136436                  CLR     R5                      ; NIB.PTR                               7366
054700  005005                  16$:    MOV     R5,R4                   ; NIB.PTR,*                             7369
054702  010504                          ASR     R4
054704  006204                          ASR     R4
054706  006204                          ASR     R4
054710  006204                          ADD     #PD.TEMP,R4
054712  062704  013120                  MOV     R4,-(SP)
054716  010446                          MOV     R5,-(SP)                ; NIB.PTR,*
054720  010546                          BIC     #177770,(SP)
054722  042716  177770                  MOV     #1,-(SP)
054726  012746  000001                  CLR     -(SP)
054732  005046                          JSR     PC,BL$GT2
054734  004767  126200                  ADD     #10,SP
054740  062706  000010                  TST     R0
054744  005700                          BNE     17$
054746  001011                          MOV     R5,-(SP)                ; NIB.PTR,*
054750  010546                          MOV     R2,-(SP)                ; BG.PAT,*
054752  010246                          MOV     #14,-(SP)
054754  012746  000014                  ADD     SP,(SP)                 ; ERR.FLG,*
054760  060616                          JSR     PC,TST.LNG.WRD
054762  004767  141166                  ADD     #6,SP
054766  062706  000006          17$:    CMP     6(SP),#1                ; ERR.FLG,*                             7373
054772  026627  000006  000001          BNE     18$
055000  001001                          INC     (SP)                    ; ERR.CNT                               7374
055002  005216                  18$:    INC     R5                      ; NIB.PTR                               7366
055004  005205                          CMP     R5,#11                  ; NIB.PTR,*
055006  020527  000011                  BLE     16$
055012  003733                          INC     R3                      ; WD.CNT                                7359
055014  005203
```

```
15792                          ;ML4                                                    14-May-1981 13:48:54    TOPS
15793                          ;                                                       14-May-1981 13:05:08    PA:<
15794
15795 055016  020327  000177             CMP     R3,#177                ; WD.CNT,*
15796 055022  003673                     BLE     11$
15797 055024  021627  000012             CMP     (SP),#12               ; ERR.CNT,*                              7380
15798 055030  003437                     BLE     19$
15799 055032  104455                     TRAP    55                     ;                                        7383
15800 055034  000121                     .WORD   121
15801 055036  010672                     .WORD   INTER
15802 055040  024060                     .WORD   DUMPER
15803 055042  012746  010230             MOV     #FNC.22,-(SP)          ;                                        7384
15804 055046  012746  006436             MOV     #WRD.13,-(SP)
15805 055052  012746  007454             MOV     #PHR.4,-(SP)
15806 055056  012746  006034             MOV     #THR.FMT,-(SP)
15807 055062  012746  000004             MOV     #4,-(SP)
15808 055066  010600                     MOV     SP,R0                  ; SP,*
15809 055070  104414                     TRAP    14
15810 055072  012716  007122             MOV     #WRD.56,(SP)           ;                                        7385
15811 055076  012746  010112             MOV     #FNC.13,-(SP)
15812 055102  012746  006024             MOV     #TWO.FMT,-(SP)
15813 055106  012746  000003             MOV     #3,-(SP)
15814 055112  010600                     MOV     SP,R0                  ; SP,*
15815 055114  104414                     TRAP    14
15816 055116  104463                     TRAP    63
15817 055120  062706  000020             ADD     #20,SP                 ;                                        7380
15818 055124  000167  000606             JMP     39$                    ;                                        7382
15819 055130  016766  134436  000002 19$: MOV     LST.BLK,2(SP)          ; *,END.CNT                              7389
15820 055136  005366  000002             DEC     2(SP)                  ; END.CNT
15821 055142  005077  136176             CLR     @ML.REG+170            ;                                        7390
15822 055146  005077  136202             CLR     @ML.REG+200            ;                                        7391
15823 055152  005077  136156             CLR     @ML.REG+160            ;                                        7392
15824 055156  005001                     CLR     R1                     ; TWICE                                  7394
15825 055160  152777  000040  136026 20$: BISB    #40,@ML.REG+40         ;                                        7395
15826 055166  016705  136410             MOV     ML.DUT,R5
15827 055172  042705  177770             BIC     #177770,R5
15828 055176  142777  000007  136010     BICB    #7,@ML.REG+40
15829 055204  150577  136004             BISB    R5,@ML.REG+40
15830 055210  005016                     CLR     (SP)                   ; ERR.CNT                                7397
15831 055212  152777  000010  136054     BISB    #10,@ML.REG+120        ;                                        7398
15832 055220  004767  140614             JSR     PC,FIRST.BLK.XFER      ;                                        7399
15833 055224  012777  000061  135722     MOV     #61,@ML.REG            ;                                        7400
15834 055232  005004                     CLR     R4                     ; BLK.CNT                                7402
15835 055234  000411                     BR      23$
15836 055236  005005                 21$: CLR     R5                     ; ADRS.CNT                               7404
15837 055240  152777  000020  136026 22$: BISB    #20,@ML.REG+120        ;                                        7406
15838 055246  005205                     INC     R5                     ; ADRS.CNT                               7404
15839 055250  020527  000177             CMP     R5,#177                ; ADRS.CNT,*
15840 055254  003771                     BLE     22$
15841 055256  005204                     INC     R4                     ; BLK.CNT                                7402
15842 055260  020466  000002         23$: CMP     R4,2(SP)               ; BLK.CNT,END.CNT
15843 055264  003764                     BLE     21$
15844 055266  152777  000040  135720     BISB    #40,@ML.REG+40         ;                                        7407
15845 055274  016705  136302             MOV     ML.DUT,R5
15846 055300  042705  177770             BIC     #177770,R5
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                                                                                                        7410
                                                                                                                        7411
                                                                                                                        7412
                                                                                                ; *,$$TMP2              7413
                                                                                                ; *,$$TMP1
                                                                                                ; $$TMP
                                                                                                ; $$TMP1
                                                                                                ; $$TMP2
                                                                                                ; WD.CNT               7415
                                                                                                                        7417
                                                                                                                        7418
                                                                                                ; *,$$TMP2              7419
                                                                                                ; *,$$TMP1
                                                                                                ; $$TMP
                                                                                                ; $$TMP1
                                                                                                ; $$TMP2
                                                                                                                        7422
                                                                                                ; NIB.PTR,*            7425
                                                                                                ; NIB.PTR,*
                                                                                                ; NIB.PTR,*
                                                                                                ; BG.PAT,*
                                                                                                ; ERR.FLG,*
                                                                                                ; ERR.FLG,*            7429
```

```
                                                                                                    14-May-1981 13:48:54    TOPS
                                                                                                    14-May-1981 13:05:08    PA:<
15904                                   ;ML4
15905                                   ;
15906
15907 055560  005216                    INC     (SP)                    ; ERR.CNT                                       7430
15908 055562  005205            35$:    INC     R5                      ; NIB.PTR                                       7422
15909 055564  020527  000011            CMP     R5,#11                  ; NIB.PTR,*
15910 055570  003733                    BLE     33$
15911 055572  005203                    INC     R3                      ; WD.CNT                                        7415
15912 055574  020327  000177            CMP     R3,#177                 ; WD.CNT,*
15913 055600  003673                    BLE     28$
15914 055602  021627  000012            CMP     (SP),#12                ; ERR.CNT,*                                     7435
15915 055606  003425                    BLE     36$
15916 055610  104455                    TRAP    55                      ;                                               7438
15917 055612  000122                    .WORD   122
15918 055614  010464                    .WORD   ASYNC
15919 055616  024060                    .WORD   DUMPER
15920 055620  012746  006406            MOV     #WRD.10,-(SP)           ;                                               7439
15921 055624  012746  007054            MOV     #WRD.51,-(SP)
15922 055630  012746  007046            MOV     #WRD.50,-(SP)
15923 055634  012746  006034            MOV     #THR.FMT,-(SP)
15924 055640  012746  000004            MOV     #4,-(SP)
15925 055644  010600                    MOV     SP,R0                   ; SP,*
15926 055646  104414                    TRAP    14
15927 055650  012766  000001  000016    MOV     #1,16(SP)               ; *,DODU.FLG                                    7440
15928 055656  062706  000012            ADD     #12,SP                  ;                                               7437
15929 055662  005266  000002    36$:    INC     2(SP)                   ; END.CNT                                       7444
15930 055666  005102                    COM     R2                      ; BG.PAT                                        7445
15931 055670  005201                    INC     R1                      ; TWICE                                         7394
15932 055672  020127  000001            CMP     R1,#1                   ; TWICE,*
15933 055676  003002                    BGT     37$
15934 055700  000167  177254            JMP     20$
15935 055704  104467            37$:    TRAP    67                      ;                                               7446
15936 055706  006000                    ROR     R0
15937 055710  103002                    BHIS    38$
15938 055712  000167  176422            JMP     1$
15939 055716  026627  000004  000001    CMP     4(SP),#1                ; DODU.FLG,*                                    7450
15940 055724  001004                    BNE     39$
15941 055726  016700  135646            MOV     ML.LUN,R0               ;                                               7453
15942 055732  104451                    TRAP    51
15943 055734  104444                    TRAP    44
15944 055736  062706  000012    39$:    ADD     #12,SP                  ;                                               7295
15945 055742  000207                    RTS     PC
15946
15947                                   ; Routine Size:  390 words
15948                                   ; Maximum stack depth per invocation:  19 words
15953
15954
15958
```

```
15960                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
15961                                   ;                                                       14-May-1981 13:05:08    PA:<
15962
15963                                           .SBTTL  T31
15967 055744                            T31::
15968 055744  004767  176360            1$:     JSR     PC,$T31                 ;                               7455
15969 055750  104466                            TRAP    66
15970 055752  006000                            ROR     R0
15971 055754  103773                            BLO     1$
15972 055756  000207                            RTS     PC
15973
15974                                   ; Routine Size:  6 words
15975                                   ; Maximum stack depth per invocation:  0 words
15980
15981
15982 ;          7458  !<BLF/PAGE>
```

```
15984 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
15985 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (82)
15986
15987 ;          7459  !
15988 ;          7460  !
15989 ;          7461  BGNTST;
15990 ;          7462
15991 ;          7463  !++
15992 ;          7464  ! TEST NUMBER:  TST 32
15993 ;          7465  !
15994 ;          7466  ! TEST NAME:    ARRAY MODULE SELECTION TEST
15995 ;          7467  !
15996 ;          7468  ! TEST DESCRIPTION:
15997 ;          7469  !
15998 ;          7470  !       TEST FOR UNIQUE ARRAY MODULE
15999 ;          7471  !       SELECTION BY:
16000 ;          7472  !
16001 ;          7473  !       1.  WRITING THE RESPECTIVE ARRAY
16002 ;          7474  !           MODULE POSITION NUMBER INTO
16003 ;          7475  !           THE FIRST GOOD NIBBLE FOUND
16004 ;          7476  !           IN THE ARRAY.  DO FOR ALL
16005 ;          7477  !           PRESENT ARRAYS.
16006 ;          7478  !
16007 ;          7479  !       2.  READ THE ARRAYS FOR THEIR
16008 ;          7480  !           RESPECTIVE POSTION NUMBERS.
16009 ;          7481  !
16010 ;          7482  ! IMPLICIT INPUTS:
16011 ;          7483  !       PD_TEMP:
16012 ;          7484  !       A BIT VECTOR OF 16 BITS WHERE
16013 ;          7485  !       THE READ PROM DATA IS STORED
16014 ;          7486  !       AND ACCESSED FROM.
16015 ;          7487  !
16016 ;          7488  !       IO_BUF :
16017 ;          7489  !       A VECTOR OF 256 WORDS WHERE
16018 ;          7490  !       DATA FOR MBUS READS AND WRITE
16019 ;          7491  !       FUNCTION ARE FOUND.
16020 ;          7492  !
16021 ;          7493  !--
16022 ;          7494
16023 ;          7495  local
16024 ;          7496      WRDS_TSTED,                          !WORDS TESTED
16025 ;          7497      ARR_SEL,                             !ARRAY SELECT
16026 ;          7498      FND_GD_NIB,                          !FOUND GOOD NIBBLE
16027 ;          7499      ARR_NUM;                             !ARRAY NUMBER
16028 ;          7500
16029 ;          7501  ARR_SEL = ZEROES;
16030 ;          7502  ARR_SEL = .ARR_SEL - .ARR_INC;           !START ARR_SEL BACK ONE ARRAY
16031 ;          7503
16032 ;          7504  incr ARR_CNT from 0 to .OP_NUM_ARR do    !TEST ALL PRESENT ARRAYS
16033 ;          7505      begin
16034 ;          7506      CLR_MBUS;
16035 ;          7507      FND_GD_NIB = ZERO;
16036 ;          7508      WRDS_TSTED = ZERO;
16037 ;          7509      DAT_DM = ONE;                        !SET DATA DIAG MODE
16038 ;          7510      MLWC = not 255;                      !LOAD WORD COUNT
```

```
16040 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
16041 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (82)
16042
16043 ;          7511      MLBA = IO_BUF;                     !LOAD IO_BUF
16044 ;          7512      ARR_SEL = .ARR_SEL + .ARR_INC;     !LOOK AT NEXT ARRAY
16045 ;          7513      MLDA = .ARR_SEL;                   !LOAD DSA WITH SELECTED ARRAY AT BLOCK ZERO
16046 ;          7514      MLCS1 = write;                     !DO A WRITE FUNCTION
16047 ;          7515
16048 ;          7516      do                                 !THIS LOOP WRITES ARRAY NUMBERS TO THE ARRAYS
16049 ;          7517          begin
16050 ;          7518          DELAY (ONE_US);
16051 ;          7519          PD_TEMP = .MLPD;               !GET THE PROM DATA
16052 ;          7520          WRDS_TSTED = .WRDS_TSTED + 1;  !COUNT WORDS TESTED
16053 ;          7521
16054 ;          7522          incr CNT from 0 to 8 do        !LOOK AT 9 NIBBLES
16055 ;          7523
16056 ;          7524              if .PD_TEMP [.CNT] IS_NOT_SET   !FIND A GOOD NIBBLE
16057 ;          7525              then
16058 ;          7526                  begin
16059 ;          7527
16060 ;          7528                  case .CNT from 0 to 8 of    !SELECT AND LOAD GOOD NIBBLE WITH ARRAY CNT
16061 ;          7529                      set
16062 ;          7530
16063 ;          7531                      [0] :
16064 ;          7532                          (MLD1)<0, 4> = .ARR_CNT;      !NIBBLE 0
16065 ;          7533
16066 ;          7534                      [1] :
16067 ;          7535                          (MLD1)<4, 4> = .ARR_CNT;      !NIBBLE 1
16068 ;          7536
16069 ;          7537                      [2] :
16070 ;          7538                          (MLD1)<8, 4> = .ARR_CNT;      !NIBBLE 2
16071 ;          7539
16072 ;          7540                      [3] :
16073 ;          7541                          (MLD1)<12, 4> = .ARR_CNT;     !NIBBLE 3
16074 ;          7542
16075 ;          7543                      [4] :
16076 ;          7544                          (MLD2)<0, 4> = .ARR_CNT;      !NIBBLE 4
16077 ;          7545
16078 ;          7546                      [5] :
16079 ;          7547                          (MLD2)<4, 4> = .ARR_CNT;      !NIBBLE 5
16080 ;          7548
16081 ;          7549                      [6] :
16082 ;          7550                          (MLD2)<8, 4> = .ARR_CNT;      !NIBBLE 6
16083 ;          7551
16084 ;          7552                      [7] :
16085 ;          7553                          (MLD2)<12, 4> = .ARR_CNT;     !NIBBLE 7
16086 ;          7554
16087 ;          7555                      [8] :
16088 ;          7556                          (MLE2)<8, 4> = .ARR_CNT;      !NIBBLE 8
16089 ;          7557                  tes;
16090 ;          7558
16091 ;          7559              FND_GD_NIB = ONE;              !SET FOUND GOOD NIBBLE FLG
16092 ;          7560              exitloop;                      !EXIT THE LOOP
16093 ;          7561              end;
16094 ;          7562
```

```
16096 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
16097 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (82)
16098
16099 ;          7563          DAT_CLK = ONE;                 !CLOCK GOOD NIBBLE INTO MEMORY & GET NXT PROM WRD
16100 ;          7564          end
16101 ;          7565      until (.FND_GD_NIB IS_SET ) or (.WRDS_TSTED eql 14);        !DO UNTIL FOUND GOOD NIBBLE OR 14 WRDS TSTED
16102 ;          7566
16103 ;          7567      if .WRDS_TSTED eql 14          !IF 14 WORDS TSTED
16104 ;          7568      then
16105 ;          7569          begin                      !THEN ERROR AND EXIT TESTED
16106 ;          7570          ERRDF (111, INTER, DUMPER);
16107 ;          7571          PRINTB (SEV_FMT, WRD_14, PHR_10, FNC_15, WRD_12, FNC_17, WRD_37, WRD_56);
16108 ;          7572          EXIT_TST;
16109 ;          7573          end;
16110 ;          7574
16111 ;          7575      end;
16112 ;          7576
16113 ;          7577  ARR_SEL = ZEROES;
16114 ;          7578  ARR_SEL = .ARR_SEL - .ARR_INC;     !START ARR SEL BACK ONE AGAIN
16115 ;          7579
16116 ;          7580  incr ARR_CNT from 0 to .OP_NUM_ARR do          !TEST ALL PRESENT ARRAYS
16117 ;          7581      begin
16118 ;          7582      BGNSUB;
16119 ;          7583      CLR_MBUS;
16120 ;          7584      FND_GD_NIB = ZERO;
16121 ;          7585      DAT_DM = ONE;                  !SET DATA DIAG MODE
16122 ;          7586      MLWC = not 255;                !LOAD WORD COUNT
16123 ;          7587      MLBA = IO_BUF;                 !LOAD UBUS ADDRESS
16124 ;          7588      ARR_SEL = .ARR_SEL + .ARR_INC; !LOOK AT NEXT ARRAY
16125 ;          7589      MLDA = .ARR_SEL;               !LOAD DSA WITH ARRAY SELECT
16126 ;          7590      MLCS1 = read;                  !DO A READ FUNCTION
16127 ;          7591
16128 ;          7592      do                             !THIS LOOP READS ARRAYS FOR ARRAY NUMBERS
16129 ;          7593          begin
16130 ;          7594          DELAY (ONE_US);
16131 ;          7595          PD_TEMP = .MLPD;           !GET THE PROM DATA
16132 ;          7596
16133 ;          7597          incr CNT from 0 to 8 do    !LOOK AT 9 NIBBLES
16134 ;          7598
16135 ;          7599              if .PD_TEMP [.CNT] IS_NOT_SET      !FIND THE GOOD NIBBLES WHERE ARR NUM IS STORED
16136 ;          7600              then
16137 ;          7601                  begin
16138 ;          7602                  DAT_CLK = ONE;     !CLOCK ARRAY WORD OUT
16139 ;          7603
16140 ;          7604                  case .CNT from 0 to 8 of       !SELECT AND READ GOOD NIBBLE
16141 ;          7605                      set
16142 ;          7606
16143 ;          7607                      [0] :
16144 ;          7608                          ARR_NUM = .(MLD1)<0, 4>;        !NIBBLE 0
16145 ;          7609
16146 ;          7610                      [1] :
16147 ;          7611                          ARR_NUM = .(MLD1)<4, 4>;        !NIBBLE 1
16148 ;          7612
16149 ;          7613                      [2] :
16150 ;          7614                          ARR_NUM = .(MLD1)<8, 4>;        !NIBBLE 2
```

```
                                                        14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
                                                        14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (82)

16152 ;ML4
16153 ;
16154
16155 ;          7615
16156 ;          7616                    [3] :
16157 ;          7617                        ARR_NUM = .(MLD1)<12, 4>;       !NIBBLE 3
16158 ;          7618
16159 ;          7619                    [4] :
16160 ;          7620                        ARR_NUM = .(MLD2)<0, 4>;        !NIBBLE 4
16161 ;          7621
16162 ;          7622                    [5] :
16163 ;          7623                        ARR_NUM = .(MLD2)<4, 4>;        !NIBBLE 5
16164 ;          7624
16165 ;          7625                    [6] :
16166 ;          7626                        ARR_NUM = .(MLD2)<8, 4>;        !NIBBLE 6
16167 ;          7627
16168 ;          7628                    [7] :
16169 ;          7629                        ARR_NUM = .(MLD2)<12, 4>;       !NIBBLE 7
16170 ;          7630
16171 ;          7631                    [8] :
16172 ;          7632                        ARR_NUM = .(MLE2)<8, 4>;        !NIBBLE 8
16173 ;          7633                    tes;
16174 ;          7634
16175 ;          7635                FND_GD_NIB = ONE;                   !SET FND GD NIB FLG
16176 ;          7636                exitloop;                           !EXIT LOOP
16177 ;          7637                end;
16178 ;          7638
16179 ;          7639            DAT_CLK = ONE;                          !CLOCK OUT NEXT PROM LOCATION
16180 ;          7640            end
16181 ;          7641        until .FND_GD_NIB IS_SET;                   !REPEAT UNTIL FOUND THE GOOD NIBBLE
16182 ;          7642
16183 ;          7643        if .ARR_CNT neq .ARR_NUM                    !SEE IF ARRAY CONTAINS IT'S ARR NUM
16184 ;          7644        then
16185 ;          7645            begin                                   !ERROR IF NOT THERE
16186 ;          7646            ERRDF (83, ASYNC, DUMPER);
16187 ;          7647            PRINTB (FOR_FMT, WRD_39, FNC_17, WRD_37, WRD_10);
16188 ;          7648            PRINTB (FMT_14, .ARR_CNT, .ARR_NUM);
16189 ;          7649            end;
16190 ;          7650
16191 ;          7651        ENDSUB;
16192 ;          7652        end;
16193 ;          7653
16194 ;          7654    ENDTST;
16198
16199                                        .SBTTL  $T32
16203 055760  004167  126132          $T32: JSR     R1,$SAVE5                 ;                                   7457
16204 055764  162706  000006                SUB     #6,SP
16205 055770  005005                        CLR     R5                        ; ARR.SEL                           7501
16206 055772  166705  133564                SUB     ARR.INC,R5                ; *,ARR.SEL                         7502
```

```
16208                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
16209                                   ;                                                       14-May-1981 13:05:08    PA:<
16210
16211 055776  016766  133556  000002          MOV     OP.NUM.ARR,2(SP)        ;                                       7504
16212 056004  005003                          CLR     R3                      ; ARR.CNT
16213 056006  000167  000702                  JMP     27$
16214 056012  152777  000040  135174  1$:     BISB    #40,@ML.REG+40          ;                                       7505
16215 056020  016702  135556                  MOV     ML.DUT,R2
16216 056024  042702  177770                  BIC     #177770,R2
16217 056030  142777  000007  135156          BICB    #7,@ML.REG+40
16218 056036  150277  135152                  BISB    R2,@ML.REG+40
16219 056042  005016                          CLR     (SP)                    ; FND.GD.NIB                            7507
16220 056044  005004                          CLR     R4                      ; WRDS.TSTED                            7508
16221 056046  152777  000010  135220          BISB    #10,@ML.REG+120         ;                                       7509
16222 056054  012777  177400  135102          MOV     #-400,@ML.REG+10        ;                                       7510
16223 056062  012777  011600  135104          MOV     #IO.BUF,@ML.REG+20      ;                                       7511
16224 056070  066705  133466                  ADD     ARR.INC,R5              ; *,ARR.SEL                             7512
16225 056074  010577  135104                  MOV     R5,@ML.REG+30           ; ARR.SEL,*                             7513
16226 056100  012777  000061  135046          MOV     #61,@ML.REG             ;                                       7514
16227 056106  012701  000001          2$:     MOV     #1,R1                   ; *,$$TMP2                              7518
16228 056112  001411                  3$:     BEQ     6$
16229 056114  016702  123776                  MOV     L$DLY,R2                ; *,$$TMP1
16230 056120  001404                          BEQ     5$
16231 056122  005066  000004          4$:     CLR     4(SP)                   ; $$TMP
16232 056126  005302                          DEC     R2                      ; $$TMP1
16233 056130  001374                          BNE     4$
16234 056132  005301                  5$:     DEC     R1                      ; $$TMP2
16235 056134  000766                          BR      3$
16236 056136  017767  135242  134754  6$:     MOV     @ML.REG+230,PD.TEMP     ;                                       7519
16237 056144  005204                          INC     R4                      ; WRDS.TSTED                            7520
16238 056146  005002                          CLR     R2                      ; CNT                                   7522
16239 056150  010201                  7$:     MOV     R2,R1                   ; CNT,*                                 7524
16240 056152  006201                          ASR     R1
16241 056154  006201                          ASR     R1
16242 056156  006201                          ASR     R1
16243 056160  062701  013120                  ADD     #PD.TEMP,R1
16244 056164  010146                          MOV     R1,-(SP)
16245 056166  010246                          MOV     R2,-(SP)                ; CNT,*
16246 056170  042716  177770                  BIC     #177770,(SP)
16247 056174  012746  000001                  MOV     #1,-(SP)
16248 056200  005046                          CLR     -(SP)
16249 056202  004767  124732                  JSR     PC,BL$GT2
16250 056206  062706  000010                  ADD     #10,SP
16251 056212  005700                          TST     R0
16252 056214  001155                          BNE     23$
16253 056216  010201                          MOV     R2,R1                   ; CNT,*                                 7528
16254 056220  006301                          ASL     R1
16255 056222  066107  056226                  ADD     8$(R1),PC
16256 056226  000022                  8$:     .WORD   9$-8$
16257 056230  000040                          .WORD   10$-8$
16258 056232  000072                          .WORD   12$-8$
16259 056234  000112                          .WORD   13$-8$
16260 056236  000146                          .WORD   15$-8$
16261 056240  000164                          .WORD   16$-8$
16262 056242  000216                          .WORD   18$-8$
```

```
16264                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
16265                                   ;                                                       14-May-1981 13:05:08    PA:<
16266
16267 056244  000236                            .WORD   19$-8$
16268 056246  000272                            .WORD   21$-8$
16269 056250  010301                    9$:     MOV     R3,R1                   ; ARR.CNT,*                             7532
16270 056252  042701  177760                    BIC     #177760,R1
16271 056256  142777  000017  135060            BICB    #17,@ML.REG+170
16272 056264  000412                            BR      11$
16273 056266  010301                    10$:    MOV     R3,R1                   ; ARR.CNT,*                             7535
16274 056270  006301                            ASL     R1
16275 056272  006301                            ASL     R1
16276 056274  006301                            ASL     R1
16277 056276  006301                            ASL     R1
16278 056300  042701  177417                    BIC     #177417,R1
16279 056304  142777  000360  135032            BICB    #360,@ML.REG+170
16280 056312  150177  135026            11$:    BISB    R1,@ML.REG+170
16281 056316  000511                            BR      22$                     ;                                       7528
16282 056320  010301                    12$:    MOV     R3,R1                   ; ARR.CNT,*                             7538
16283 056322  000301                            SWAB    R1
16284 056324  042701  170377                    BIC     #170377,R1
16285 056330  042777  007400  135006            BIC     #7400,@ML.REG+170
16286 056336  000413                            BR      14$
16287 056340  010301                    13$:    MOV     R3,R1                   ; ARR.CNT,*                             7541
16288 056342  000301                            SWAB    R1
16289 056344  006301                            ASL     R1
16290 056346  006301                            ASL     R1
16291 056350  006301                            ASL     R1
16292 056352  006301                            ASL     R1
16293 056354  042701  007777                    BIC     #7777,R1
16294 056360  042777  170000  134756            BIC     #170000,@ML.REG+170
16295 056366  050177  134752            14$:    BIS     R1,@ML.REG+170
16296 056372  000463                            BR      22$                     ;                                       7528
16297 056374  010301                    15$:    MOV     R3,R1                   ; ARR.CNT,*                             7544
16298 056376  042701  177760                    BIC     #177760,R1
16299 056402  142777  000017  134744            BICB    #17,@ML.REG+200
16300 056410  000412                            BR      17$
16301 056412  010301                    16$:    MOV     R3,R1                   ; ARR.CNT,*                             7547
16302 056414  006301                            ASL     R1
16303 056416  006301                            ASL     R1
16304 056420  006301                            ASL     R1
16305 056422  006301                            ASL     R1
16306 056424  042701  177417                    BIC     #177417,R1
16307 056430  142777  000360  134716            BICB    #360,@ML.REG+200
16308 056436  150177  134712            17$:    BISB    R1,@ML.REG+200
16309 056442  000437                            BR      22$                     ;                                       7528
16310 056444  010301                    18$:    MOV     R3,R1                   ; ARR CNT,*                             7550
16311 056446  000301                            SWAB    R1
16312 056450  042701  170377                    BIC     #170377,R1
16313 056454  042777  007400  134672            BIC     #7400,@ML.REG+200
16314 056462  000413                            BR      20$
16315 056464  010301                    19$:    MOV     R3,R1                   ; ARR.CNT,*                             7553
16316 056466  000301                            SWAB    R1
16317 056470  006301                            ASL     R1
16318 056472  006301                            ASL     R1
```

```
                                                                                                                14-May-1981 13:48:54    TOPS
                                                                                                                14-May-1981 13:05:08    PA:<
16320                                   ;ML4
16321                                   ;
16322
16323 056474  006301                            ASL     R1
16324 056476  006301                            ASL     R1
16325 056500  042701  007777                    BIC     #7777,R1
16326 056504  042777  170000  134642            BIC     #170000,@ML.REG+200
16327 056512  050177  134636           20$:     BIS     R1,@ML.REG+200
16328 056516  000411                            BR      22$                     ;                                               7528
16329 056520  010301                    21$:    MOV     R3,R1                   ; ARR.CNT,*                                     7556
16330 056522  000301                            SWAB    R1
16331 056524  042701  170377                    BIC     #170377,R1
16332 056530  042777  007400  134576            BIC     #7400,@ML.REG+160
16333 056536  050177  134572                    BIS     R1,@ML.REG+160
16334 056542  012716  000001           22$:     MOV     #1,(SP)                 ; *,FND.GD.NIB                                  7559
16335 056546  000406                            BR      24$                     ;                                               7560
16336 056550  005202                    23$:    INC     R2                      ; CNT                                           7522
16337 056552  020227  000010                    CMP     R2,#10                  ; CNT,*
16338 056556  003002                            BGT     24$
16339 056560  000167  177364                    JMP     7$
16340 056564  152777  000020  134502   24$:     BISB    #20,@ML.REG+120         ;                                               7563
16341 056572  021627  000001                    CMP     (SP),#1                 ; FND.GD.NIB,*                                  7565
16342 056576  001405                            BEQ     25$
16343 056600  020427  000016                    CMP     R4,#16                  ; WRDS.TSTED,*
16344 056604  001402                            BEQ     25$
16345 056606  000167  177274                    JMP     2$
16346 056612  020427  000016           25$:     CMP     R4,#16                  ; WRDS.TSTED,*                                  7567
16347 056616  001035                            BNE     26$
16348 056620  104455                            TRAP    55                      ;                                               7570
16349 056622  000157                            .WORD   157
16350 056624  010672                            .WORD   INTER
16351 056626  024060                            .WORD   DUMPER
16352 056630  012746  007122                    MOV     #WRD.56,-(SP)           ;                                               7571
16353 056634  012746  006700                    MOV     #WRD.37,-(SP)
16354 056640  012746  010162                    MOV     #FNC.17,-(SP)
16355 056644  012746  006426                    MOV     #WRD.12,-(SP)
16356 056650  012746  010136                    MOV     #FNC.15,-(SP)
16357 056654  012746  007564                    MOV     #PHR.10,-(SP)
16358 056660  012746  006442                    MOV     #WRD.14,-(SP)
16359 056664  012746  006120                    MOV     #SEV.FMT,-(SP)
16360 056670  012746  000010                    MOV     #10,-(SP)
16361 056674  010600                            MOV     SP,R0                   ; SP,*
16362 056676  104414                            TRAP    14
16363 056700  104463                            TRAP    63
16364 056702  062706  000022                    ADD     #22,SP                  ;                                               7567
16365 056706  000167  000562                    JMP     56$                     ;                                               7569
16366 056712  005203                    26$:    INC     R3                      ; ARR.CNT                                       7504
16367 056714  020366  000002           27$:     CMP     R3,2(SP)                ; ARR.CNT,*
16368 056720  003002                            BGT     28$
16369 056722  000167  177064                    JMP     1$
16370 056726  005005                    28$:    CLR     R5                      ; ARR.SEL                                       7577
16371 056730  166705  132626                    SUB     ARR.INC,R5              ; *,ARR.SEL                                     7578
16372 056734  016766  132620  000002            MOV     OP.NUM.ARR,2(SP)        ;                                               7580
16373 056742  005004                            CLR     R4                      ; ARR.CNT
16374 056744  000167  000516                    JMP     55$
```

```
                                                                                                    14-May-1981 13:48:54    TOPS
                                                                                                    14-May-1981 13:05:08    PA:<
16376                                ;ML4
16377                                ;
16378
16379 056750  104402                29$:    TRAP    2                                                                       7581
16380 056752  152777  000040  134234        BISB    #40,@ML.REG+40                                                          7582
16381 056760  016702  134616              MOV     ML.DUT,R2
16382 056764  042702  177770              BIC     #177770,R2
16383 056770  142777  000007  134216        BICB    #7,@ML.REG+40
16384 056776  150277  134212              BISB    R2,@ML.REG+40
16385 057002  005016                      CLR     (SP)                    ; FND.GD.NIB                                    7584
16386 057004  152777  000010  134262        BISB    #10,@ML.REG+120                                                         7585
16387 057012  012777  177400  134144        MOV     #-400,@ML.REG+10                                                        7586
16388 057020  012777  011600  134146        MOV     #IO.BUF,@ML.REG+20                                                      7587
16389 057026  066705  132530              ADD     ARR.INC,R5              ; *,ARR.SEL                                     7588
16390 057032  010577  134146              MOV     R5,@ML.REG+30           ; ARR.SEL,*                                     7589
16391 057036  012777  000071  134110        MOV     #71,@ML.REG                                                             7590
16392 057044  012701  000001        30$:    MOV     #1,R1                   ; *,$$TMP2                                      7594
16393 057050  001411                31$:    BEQ     34$
16394 057052  016702  123040              MOV     L$DLY,R2                ; *,$$TMP1
16395 057056  001404                      BEQ     33$
16396 057060  005066  000004        32$:    CLR     4(SP)                   ; $$TMP
16397 057064  005302                      DEC     R2                      ; $$TMP1
16398 057066  001374                      BNE     32$
16399 057070  005301                33$:    DEC     R1                      ; $$TMP2
16400 057072  000766                      BR      31$
16401 057074  017767  134304  134016  34$:    MOV     @ML.REG+230,PD.TEMP                                                     7595
16402 057102  005002                      CLR     R2                      ; CNT                                           7597
16403 057104  010201                35$:    MOV     R2,R1                   ; CNT,*                                         7599
16404 057106  006201                      ASR     R1
16405 057110  006201                      ASR     R1
16406 057112  006201                      ASR     R1
16407 057114  062701  013120              ADD     #PD.TEMP,R1
16408 057120  010146                      MOV     R1,-(SP)
16409 057122  010246                      MOV     R2,-(SP)                ; CNT,*
16410 057124  042716  177770              BIC     #177770,(SP)
16411 057130  012746  000001              MOV     #1,-(SP)
16412 057134  005046                      CLR     -(SP)
16413 057136  004767  123776              JSR     PC,BL$GT2
16414 057142  062706  000010              ADD     #10,SP
16415 057146  005700                      TST     R0
16416 057150  001070                      BNE     50$
16417 057152  152777  000020  134114        BISB    #20,@ML.REG+120                                                         7602
16418 057160  010201                      MOV     R2,R1                   ; CNT,*                                         7604
16419 057162  006301                      ASL     R1
16420 057164  066107  057170              ADD     36$(R1),PC
16421 057170  000022                36$:    .WORD   37$-36$
16422 057172  000030                      .WORD   38$-36$
16423 057174  000036                      .WORD   39$-36$
16424 057176  000044                      .WORD   40$-36$
16425 057200  000052                      .WORD   41$-36$
16426 057202  000060                      .WORD   42$-36$
16427 057204  000076                      .WORD   44$-36$
16428 057206  000104                      .WORD   45$-36$
16429 057210  000122                      .WORD   47$-36$
16430 057212  117703  134126        37$:    MOVB    @ML.REG+170,R3          ; *,ARR.NUM                                     7608
```

```
                                                                                                14-May-1981 13:48:54    TOPS
                                                                                                14-May-1981 13:05:08    PA:<
 16432                          ;ML4
 16433                          ;
 16434
 16435 057216  000440                    BR      49$
 16436 057220  117703  134120    38$:    MOVB    @ML.REG+170,R3          ; *,ARR.NUM                             7611
 16437 057224  000413                    BR      43$
 16438 057226  017703  134112    39$:    MOV     @ML.REG+170,R3          ; *,ARR.NUM                             7614
 16439 057232  000431                    BR      48$
 16440 057234  017703  134104    40$:    MOV     @ML.REG+170,R3          ; *,ARR.NUM                             7617
 16441 057240  000417                    BR      46$
 16442 057242  117703  134106    41$:    MOVB    @ML.REG+200,R3          ; *,ARR.NUM                             7620
 16443 057246  000424                    BR      49$
 16444 057250  117703  134100    42$:    MOVB    @ML.REG+200,R3          ; *,ARR.NUM                             7623
 16445 057254  006203            43$:    ASR     R3                      ; ARR.NUM
 16446 057256  006203                    ASR     R3                      ; ARR.NUM
 16447 057260  006203                    ASR     R3                      ; ARR.NUM
 16448 057262  006203                    ASR     R3                      ; ARR.NUM
 16449 057264  000415                    BR      49$
 16450 057266  017703  134062    44$:    MOV     @ML.REG+200,R3          ; *,ARR.NUM                             7626
 16451 057272  000411                    BR      48$
 16452 057274  017703  134054    45$:    MOV     @ML.REG+200,R3          ; *,ARR.NUM                             7629
 16453 057300  006203            46$:    ASR     R3                      ; ARR.NUM
 16454 057302  006203                    ASR     R3                      ; ARR.NUM
 16455 057304  006203                    ASR     R3                      ; ARR.NUM
 16456 057306  006203                    ASR     R3                      ; ARR.NUM
 16457 057310  000402                    BR      48$
 16458 057312  017703  134016    47$:    MOV     @ML.REG+160,R3          ; *,ARR.NUM                             7632
 16459 057316  000303            48$:    SWAB    R3                      ; ARR.NUM
 16460 057320  042703  177760    49$:    BIC     #177760,R3              ; *,ARR.NUM
 16461 057324  012716  000001            MOV     #1,(SP)                 ; *,FND.GD.NIB                          7635
 16462 057330  000404                    BR      51$                     ;                                       7636
 16463 057332  005202            50$:    INC     R2                      ; CNT                                   7597
 16464 057334  020227  000010            CMP     R2,#10                  ; CNT,*
 16465 057340  003661                    BLE     35$
 16466 057342  152777  000020  133724 51$:    BISB    #20,@ML.REG+120         ;                               7639
 16467 057350  021627  000001            CMP     (SP),#1                 ; FND.GD.NIB,*                          7641
 16468 057354  001233                    BNE     30$
 16469 057356  020403                    CMP     R4,R3                   ; ARR.CNT,ARR.NUM                       7643
 16470 057360  001434                    BEQ     52$
 16471 057362  104455                    TRAP    55                      ;                                       7646
 16472 057364  000123                    .WORD   123
 16473 057366  010464                    .WORD   ASYNC
 16474 057370  024060                    .WORD   DUMPER
 16475 057372  012746  006406            MOV     #WRD.10,-(SP)           ;                                       7647
 16476 057376  012746  006700            MOV     #WRD.37,-(SP)
 16477 057402  012746  010162            MOV     #FNC.17,-(SP)
 16478 057406  012746  006716            MOV     #WRD.39,-(SP)
 16479 057412  012746  006046            MOV     #FOR.FMT,-(SP)
 16480 057416  012746  000005            MOV     #5,-(SP)
 16481 057422  010600                    MOV     SP,R0                   ; SP,*
 16482 057424  104414                    TRAP    14
 16483 057426  010316                    MOV     R3,(SP)                 ; ARR.NUM,*                             7648
 16484 057430  010446                    MOV     R4,-(SP)                ; ARR.CNT,*
 16485 057432  012746  005110            MOV     #FMT.14,-(SP)
 16486 057436  012746  000003            MOV     #3,-(SP)
```

```
                                ;ML4                                                    14-May-1981 13:48:54    TOPS
                                ;                                                       14-May-1981 13:05:08    PA:<
                                
057442  010600                          MOV     SP,R0                   ; SP,*
057444  104414                          TRAP    14
057446  062706  000022                  ADD     #22,SP                  ;                               7645
057452  104467                  52$:    TRAP    67                      ;                               7649
057454  006000                          ROR     R0
057456  103002                          BHIS    54$
057460  000167  177264          53$:    JMP     29$
057464  005204                  54$:    INC     R4                      ; ARR.CNT                       7580
057466  020466  000002          55$:    CMP     R4,2(SP)                ; ARR.CNT,*
057472  003772                          BLE     53$
057474  062706  000006          56$:    ADD     #6,SP                   ;                               7457
057500  000207                          RTS     PC

                                ; Routine Size:  425 words
                                ; Maximum stack depth per invocation:  18 words


                                        .SBTTL  T32
057502                          T32::
057502  004767  176252          1$:     JSR     PC,$T32                 ;                               7652
057506  104466                          TRAP    66
057510  006000                          ROR     R0
057512  103773                          BLO     1$
057514  000207                          RTS     PC

                                ; Routine Size:  6 words
                                ; Maximum stack depth per invocation:  0 words


;               7655    !<BLF/PAGE>
```

```
;ML4
;
                7656  !
                7657  BGNTST;
                7658
                7659  !++
                7660  ! TEST NUMBER:  TST 33
                7661  !
                7662  ! TEST NAME:    SEQUENCER EXISTENCE TEST
                7663  !
                7664  ! TEST DESCRIPTION:
                7665  !
                7666  !       TEST SEQUENCER TIMING AND CONTROL
                7667  !       LOGIC FOR EXISTENCE BY:
                7668  !
                7669  !       1.  FIRST WRITING THE GOOD BLOCK VIA DAT_DM MODE WITH A BACKGROUND
                7670  !           PATTERN OF ONES.
                7671  !
                7672  !
                7673  !       2.  VIA A MBUS WRITE FUNCTION WRITE ONES INTO THE GOOD BLOCK.
                7674  !
                7675  !       3.  THEN VIA DAT_DM READ GOOD NIBBLES IN THE GOOD BLOCK AND XOR THEM
                7676  !           AGAINST THE BACKGROUND PATTERN.
                7677  !
                7678  !           RECORD THE NUMBER OF NIBBLES THAT WERE UNCHANGED OR PARTIALLY CHANGED
                7679  !           BY THE MBUS WIRTE.
                7680  !
                7681  !       4.  REPORT A FATAL ERROR AND DROP THIS UNIT IF THE NIBBLES TESTED EQUALS
                7682  !           THE COUNT OF UNCHANGED NIBBLES.
                7683  !
                7684  !           REPORT AN INTERMEDIATE DIAGNOSTIC MESSAGE IF AT LEAST SOME NIBBLES
                7685  !           WERE CHANGED BY THE MBUS WRITE.
                7686  !
                7687  ! IMPLICIT INPUTS:
                7688  !       PD_TEMP:
                7689  !       A BIT VECTOR OF 16 BITS WHERE THE READ PROM DATA IS STORED AND ACCESSED FROM.
                7690  !
                7691  !       IO_BUF:
                7692  !       A VECTOR OF 256 WORDS WHERE DATA FOR MBUS READS AND WRITE FUNCTIONS
                7693  !       ARE FOUND.
                7694  !--
                7695
                7696  local
                7697      BG_PAT,                         !BACK GROUND PATTERN
                7698      SUM_BAD,                        !SUM NIBBLE ARE BAD
                7699      NIB_TSTED,                      !NIBBLES TESTED
                7700      ALL_BAD,                        !ALL NIBBLES ARE BAD
                7701      RESULT;                         !RESULTS OF XOR
                7702
                7703  CLR_MBUS;
                7704  BG_PAT = ONES;                      !BACKGROUND EQL ONES
                7705  MLD1 = .BG_PAT;                     !LOAD DATA DIAG REG WITH BG PAT
                7706  MLD2 = .BG_PAT;
                7707  MLE2 = .BG_PAT;
;ML4
;
```

```
16594
16595 ;          7708  DAT_DM_XFER ();                          !SET UP A  DATA DIAG XFERR AT THE GOOD BLOCK
16596 ;          7709  MLCS1 = write;                           !DO A WRITE FUNCTION
16597 ;          7710
16598 ;          7711  incr WD_CNT from 0 to 127 do             !WRITE BLOCK WITH BG PAT
16599 ;          7712      begin
16600 ;          7713      DELAY (ONE_US);
16601 ;          7714      DAT_CLK = ONE;
16602 ;          7715      end;
16603 ;          7716
16604 ;          7717  CLR_MBUS;
16605 ;          7718  IO_BUF = ZEROES;                         !LOAD FIRST WORD OF IO_BUF
16606 ;          7719  BAI = ONE;                               !MAKE XFERR SET ON ONE ADRS
16607 ;          7720  GD_BLK_XFER ();                          !SET UP A GOOD BLOCK XFERR
16608 ;          7721  MLCS1 = write;                           !DO A WRITE FUNCTION
16609 ;          7722  TIME_OUT_LOOP;
16610 ;          7723  BGNSUB;
16611 ;          7724  CLR_MBUS;
16612 ;          7725  ALL_BAD = ZEROES;
16613 ;          7726  SUM_BAD = ZEROES;
16614 ;          7727  NIB_TSTED = ZEROES;
16615 ;          7728  DAT_DM_XFER ();                          !SET UP SAME DATA DIAG XFERR
16616 ;          7729  MLCS1 = read;                            !DO A READ FUNCTION
16617 ;          7730  DELAY (ONE_US);
16618 ;          7731
16619 ;          7732  incr WD_CNT from 0 to 112 do             !READ 113 LONG WORDS
16620 ;          7733      begin
16621 ;          7734      PD_TEMP = .MLPD;                     !GET PROM DATA
16622 ;          7735      DAT_CLK = ONE;                       !CLOCK OUT THE DATA WORD
16623 ;          7736      DELAY (ONE_US);
16624 ;          7737      RD_LNG_WRD;                          !READ THE DATA WORD
16625 ;          7738
16626 ;          7739      incr NIB_PTR from 0 to 8 do          !LOOK AT 9 NIBBLES
16627 ;          7740
16628 ;          7741          if .PD_TEMP [.NIB_PTR] eql ZERO  !FIND GOOD NIBBLE
16629 ;          7742          then
16630 ;          7743              begin
16631 ;          7744              NIB_TSTED = .NIB_TSTED + 1;  !INCREMENT COUNT OF NIBBLES TESTED
16632 ;          7745              XOR_LNG_WRD (.NIB_PTR, .BG_PAT, RESULT);    !XOR NIBBLE WITH BG PAT
16633 ;          7746
16634 ;          7747              if .RESULT<0, 4> eql ZERO    !SEE IF ALL BITS IN NIBBLE WERE BAD?
16635 ;          7748              then
16636 ;          7749                  ALL_BAD = .ALL_BAD + 1   !INCREMENT COUNT IF ALL BAD
16637 ;          7750              else
16638 ;          7751
16639 ;          7752                  if .RESULT<0, 4> neq %o'17' then SUM_BAD = .SUM_BAD + 1;
16640 ;          7753
16641 ;          7754                                           !SEE IF SOME BITS IN NIBBLE WERE BAD
16642 ;          7755              end;                         !INCREMENT COUNT IF SOME BAD
16643 ;          7756
16644 ;          7757      end;
16645 ;          7758
16646 ;          7759  ENDSUB;
16647 ;ML4                                                      14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
16648 ;                                                        14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (83)
16649
16650 ;          7760
```

```
  16651 ;          7761  if .NIB_TSTED eql .ALL_BAD                    !WHERE ALL NIBBLES XFERR'ED ALL BAD?
  16652 ;          7762  then                                           !ERROR IF ALL BAD
  16653 ;          7763      begin
  16654 ;          7764      ERRDF (84, SYNC, DUMPER);
  16655 ;          7765      PRINTB (SEV_FMT, WRD_22, PHR_4, WRD_9, WRD_12, WRD_23, FNC_5, WRD_19);
  16656 ;          7766      DODU (.ML_LUN);
  16657 ;          7767      DOCLN;
  16658 ;          7768      end
  16659 ;          7769  else
  16660 ;          7770
  16661 ;          7771      if .SUM_BAD gtr ZERO                       !SEE IF SEE NIBBLE WERE BAD?
  16662 ;          7772      then
  16663 ;          7773          begin                                  !SOME BAD IS OK SO GIVE INTERMEDIATE ERROR
  16664 ;          7774          ERRDF (85, INTER, DUMPER);
  16665 ;          7775          PRINTB (SIX_FMT, PHR_4, WRD_9, WRD_12, WRD_23, FNC_5, WRD_19);
  16666 ;          7776          end;
  16667 ;          7777
  16668 ;          7778  ENDTST;
  16672
  16673                                        .SBTTL  $T33
  16677 057516  004167  124374          $T33:  JSR     R1,$SAVE5                ;                                  7654
  16678 057522  162706  000010                 SUB     #10,SP
  16679 057526  152777  000040  133460         BISB    #40,@ML.REG+40           ;                                  7701
  16680 057534  016705  134042                 MOV     ML.DUT,R5
  16681 057540  042705  177770                 BIC     #177770,R5
  16682 057544  142777  000007  133442         BICB    #7,@ML.REG+40
  16683 057552  150577  133436                 BISB    R5,@ML.REG+40
  16684 057556  012766  177777  000002         MOV     #-1,2(SP)                ; *,BG.PAT                         7704
  16685 057564  012777  177777  133552         MOV     #-1,@ML.REG+170          ;                                  7705
  16686 057572  016677  000002  133554         MOV     2(SP),@ML.REG+200        ; BG.PAT,*                         7706
  16687 057600  016677  000002  133526         MOV     2(SP),@ML.REG+160        ; BG.PAT,*                         7707
  16688 057606  004767  134350                 JSR     PC,DAT.DM.XFER           ;                                  7708
  16689 057612  012777  000061  133334         MOV     #61,@ML.REG              ;                                  7709
  16690 057620  005000                 CLR     R0                               ; WD.CNT                           7711
  16691 057622  012701  000001         1$:     MOV     #1,R1                    ; *,$$TMP2                         7713
  16692 057626  001411                 2$:     BEQ     5$
  16693 057630  016702  122262                 MOV     L$DLY,R2                 ; *,$$TMP1
  16694 057634  001404                 BEQ     4$
  16695 057636  005066  000006         3$:     CLR     6(SP)                    ; $$TMP
  16696 057642  005302                 DEC     R2                               ; $$TMP1
  16697 057644  001374                 BNE     3$
  16698 057646  005301                 4$:     DEC     R1                       ; $$TMP2
  16699 057650  000766                 BR      2$
  16700 057652  152777  000020  133414  5$:     BISB    #20,@ML.REG+120          ;                                  7714
  16701 057660  005200                 INC     R0                               ; WD.CNT                           7711
  16702                                ;ML4                                                    14-May-1981 13:48:54  TOPS
  16703                                ;                                                       14-May-1981 13:05:08  PA:<
  16704
  16705 057662  020027  000177                 CMP     R0,#177                  ; WD.CNT,*
  16706 057666  003755                 BLE     1$
  16707 057670  152777  000040  133316         BISB    #40,@ML.REG+40           ;                                  7715
  16708 057676  016705  133700                 MOV     ML.DUT,R5
  16709 057702  042705  177770                 BIC     #177770,R5
  16710 057706  142777  000007  133300         BICB    #7,@ML.REG+40
  16711 057714  150577  133274                 BISB    R5,@ML.REG+40
  16712 057720  005067  131654                 CLR     IO.BUF                   ;                                  7718
  16713 057724  152777  000010  133262         BISB    #10,@ML.REG+40           ;                                  7719
```

```
16714 057732  004767  136132                    JSR     PC,GD.BLK.XFER          ;                                       7720
16715 057736  012777  000061  133210            MOV     #61,@ML.REG             ;                                       7721
16716 057744  105777  133254           6$:      TSTB    @ML.REG+50
16717 057750  100375                           BPL     6$
16718 057752  104402                    7$:      TRAP    2                       ;                                       7722
16719 057754  152777  000040  133232            BISB    #40,@ML.REG+40          ;                                       7723
16720 057762  016703  133614                    MOV     ML.DUT,R3
16721 057766  042703  177770                    BIC     #177770,R3
16722 057772  142777  000007  133214            BICB    #7,@ML.REG+40
16723 060000  150377  133210                    BISB    R3,@ML.REG+40
16724 060004  005004                            CLR     R4                      ; ALL.BAD                               7725
16725 060006  005016                            CLR     (SP)                    ; SUM.BAD                               7726
16726 060010  005005                            CLR     R5                      ; NIB.TSTED                             7727
16727 060012  004767  134144                    JSR     PC,DAT.DM.XFER          ;                                       7728
16728 060016  012777  000071  133130            MOV     #71,@ML.REG             ;                                       7729
16729 060024  012701  000001                    MOV     #1,R1                   ; *,$$TMP2                              7730
16730 060030  001411                    8$:      BEQ     11$
16731 060032  016702  122060                    MOV     L$DLY,R2                ; *,$$TMP1
16732 060036  001404                            BEQ     10$
16733 060040  005066  000006            9$:      CLR     6(SP)                   ; $$TMP
16734 060044  005302                            DEC     R2                      ; $$TMP1
16735 060046  001374                            BNE     9$
16736 060050  005301                    10$:     DEC     R1                      ; $$TMP2
16737 060052  000766                            BR      8$
16738 060054  005003                    11$:     CLR     R3                      ; WD.CNT                                7732
16739 060056  017767  133322  133034    12$:     MOV     @ML.REG+230,PD.TEMP     ;                                       7734
16740 060064  152777  000020  133202            BISB    #20,@ML.REG+120         ;                                       7735
16741 060072  012701  000001                    MOV     #1,R1                   ; *,$$TMP2                              7736
16742 060076  001411                    13$:     BEQ     16$
16743 060100  016702  122012                    MOV     L$DLY,R2                ; *,$$TMP1
16744 060104  001404                            BEQ     15$
16745 060106  005066  000006            14$:     CLR     6(SP)                   ; $$TMP
16746 060112  005302                            DEC     R2                      ; $$TMP1
16747 060114  001374                            BNE     14$
16748 060116  005301                    15$:     DEC     R1                      ; $$TMP2
16749 060120  000766                            BR      13$
16750 060122  017767  133216  131022    16$:     MOV     @ML.REG+170,D1.TEMP
16751 060130  017767  133220  131016            MOV     @ML.REG+200,D2.TEMP
16752 060136  017767  133172  131012            MOV     @ML.REG+160,E2.TEMP
16753 060144  005001                            CLR     R1                      ; NIB.PTR                                       7739
16754 060146  010102                    17$:     MOV     R1,R2                   ; NIB.PTR,*                             7741
16755 060150  006202                            ASR     R2
16756 060152  006202                            ASR     R2
16757                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
16758                                   ;                                                       14-May-1981 13:05:08    PA:<
16759
16760 060154  006202                            ASR     R2
16761 060156  062702  013120                    ADD     #PD.TEMP,R2
16762 060162  010246                            MOV     R2,-(SP)
16763 060164  010146                            MOV     R1,-(SP)                ; NIB.PTR,*
16764 060166  042716  177770                    BIC     #177770,(SP)
16765 060172  012746  000001                    MOV     #1,-(SP)
16766 060176  005046                            CLR     -(SP)
16767 060200  004767  122734                    JSR     PC,BL$GT2
16768 060204  062706  000010                    ADD     #10,SP
16769 060210  005700                            TST     R0
16770 060212  001027                            BNE     20$
```

```
16771 060214 005205                    INC     R5                      ; NIB.TSTED                     7744
16772 060216 010146                    MOV     R1,-(SP)                ; NIB.PTR,*                     7745
16773 060220 016646 000004             MOV     4(SP),-(SP)             ; BG.PAT,*
16774 060224 012746 000012             MOV     #12,-(SP)
16775 060230 060616                    ADD     SP,(SP)                 ; RESULT,*
16776 060232 004767 136412             JSR     PC,XOR.LNG.WRD
16777 060236 016602 000012             MOV     12(SP),R2               ; RESULT,*                      7747
16778 060242 042702 177760             BIC     #177760,R2
16779 060246 001002                    BNE     18$
16780 060250 005204                    INC     R4                      ; ALL.BAD                       7749
16781 060252 000405                    BR      19$                     ;                               7747
16782 060254 020227 000017      18$:   CMP     R2,#17                  ;                               7752
16783 060260 001402                    BEQ     19$
16784 060262 005266 000006             INC     6(SP)                   ; SUM.BAD
16785 060266 062706 000006      19$:   ADD     #6,SP                   ;                               7743
16786 060272 005201             20$:   INC     R1                      ; NIB.PTR                       7739
16787 060274 020127 000010             CMP     R1,#10                  ; NIB.PTR,*
16788 060300 003722                    BLE     17$
16789 060302 005203                    INC     R3                      ; WD.CNT                        7732
16790 060304 020327 000160             CMP     R3,#160                 ; WD.CNT,*
16791 060310 003662                    BLE     12$
16792 060312 104467                    TRAP    67                      ;                               7757
16793 060314 006000                    ROR     R0
16794 060316 103615                    BLO     7$
16795 060320 020504                    CMP     R5,R4                   ; NIB.TSTED,ALL.BAD             7761
16796 060322 001037                    BNE     21$
16797 060324 104455                    TRAP    55                      ;                               7764
16798 060326 000124                    .WORD   124
16799 060330 010526                    .WORD   SYNC
16800 060332 024060                    .WORD   DUMPER
16801 060334 012746 006506             MOV     #WRD.19,-(SP)           ;                               7765
16802 060340 012746 007774             MOV     #FNC.5,-(SP)
16803 060344 012746 006544             MOV     #WRD.23,-(SP)
16804 060350 012746 006426             MOV     #WRD.12,-(SP)
16805 060354 012746 006374             MOV     #WRD.9,-(SP)
16806 060360 012746 007454             MOV     #PHR.4,-(SP)
16807 060364 012746 006530             MOV     #WRD.22,-(SP)
16808 060370 012746 006120             MOV     #SEV.FMT,-(SP)
16809 060374 012746 000010             MOV     #10,-(SP)
16810 060400 010600                    MOV     SP,R0                   ; SP,*
16811 060402 104414                    TRAP    14
16812                          ;ML4                                                  14-May-1981 13:48:54    TOPS
16813                          ;                                                     14-May-1981 13:05:08    PA:<
16814
16815 060404 016700 133170             MOV     ML.LUN,R0               ;                               7766
16816 060410 104451                    TRAP    51
16817 060412 104444                    TRAP    44
16818 060414 062706 000022             ADD     #22,SP                  ;                               7763
16819 060420 000432                    BR      22$                     ;                               7761
16820 060422 005716             21$:   TST     (SP)                    ; SUM.BAD                       7771
16821 060424 003430                    BLE     22$
16822 060426 104455                    TRAP    55                      ;                               7774
16823 060430 000125                    .WORD   125
16824 060432 010672                    .WORD   INTER
16825 060434 024060                    .WORD   DUMPER
16826 060436 012746 006506             MOV     #WRD.19,-(SP)           ;                               7775
16827 060442 012746 007774             MOV     #FNC.5,-(SP)
```

```
 16828 060446  012746  006544                MOV     #WRD.23,-(SP)
 16829 060452  012746  006426                MOV     #WRD.12,-(SP)
 16830 060456  012746  006374                MOV     #WRD.9,-(SP)
 16831 060462  012746  007454                MOV     #PHR.4,-(SP)
 16832 060466  012746  006100                MOV     #SIX.FMT,-(SP)
 16833 060472  012746  000007                MOV     #7,-(SP)
 16834 060476  010600                        MOV     SP,R0                           ; SP,*
 16835 060500  104414                        TRAP    14
 16836 060502  062706  000020                ADD     #20,SP                          ;                       7773
 16837 060506  062706  000010        22$:    ADD     #10,SP                          ;                       7654
 16838 060512  000207                        RTS     PC
 16839
 16840                              ; Routine Size:  255 words
 16841                              ; Maximum stack depth per invocation:  19 words
 16846
 16847
 16851
 16852                                      .SBTTL  T33
 16856 060514                       T33::
 16857 060514  004767  176776       1$:     JSR     PC,$T33                         ;                       7776
 16858 060520  104466                        TRAP    66
 16859 060522  006000                        ROR     R0
 16860 060524  103773                        BLO     1$
 16861 060526  000207                        RTS     PC
 16862
 16863                              ; Routine Size:  6 words
 16864                              ; Maximum stack depth per invocation:  0 words
 16872
 16873
 16874 ;          7779  !<BLF/PAGE>
```

```
16876 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
16877 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (84)
16878
16879 ;          7780  !
16880 ;          7781  !
16881 ;          7782  BGNTST;
16882 ;          7783
16883 ;          7784  !++
16884 ;          7785  ! TEST NUMBER:  TST 34
16885 ;          7786  !
16886 ;          7787  ! TEST NAME:    SYNC DATA BUS CONTINUITY TEST (WRITE PATH)
16887 ;          7788  !
16888 ;          7789  ! TEST DESCRIPTION:
16889 ;          7790  !
16890 ;          7791  !       TEST THE CONTINUITY OF THE
16891 ;          7792  !       SYNCHRONOUS MODULE WRITE PATH
16892 ;          7793  !       DATA BUS BY:
16893 ;          7794  !
16894 ;          7795  !       1.  VIA DAT_DM MODE WRITE A
16895 ;          7796  !           BACKGROUND PATTERN OF ONES
16896 ;          7797  !           INTO THE GOOD BLOCK.
16897 ;          7798  !
16898 ;          7799  !       2.  VIA MBUS WRITE FUNCTION
16899 ;          7800  !           WRITE A ZEROES PATTERN INTO
16900 ;          7801  !           THE GOOD BLOCK.
16901 ;          7802  !
16902 ;          7803  !       3.  VIA DAT_DM MODE READ GOOD
16903 ;          7804  !           NIBBLES IN THE GOOD BLOCK FOR
16904 ;          7805  !           ZEROES.
16905 ;          7806  !
16906 ;          7807  !       4.  REPEAT WITH COMPLIMENT
16907 ;          7808  !           DATA AND BACKGROUND PATTERNS.
16908 ;          7809  !
16909 ;          7810  ! IMPLICIT INPUTS:
16910 ;          7811  !
16911 ;          7812  !       PD_TEMP
16912 ;          7813  !
16913 ;          7814  !       A BIT VECTOR OF 16 BITS WHERE
16914 ;          7815  !       THE READ PROM DATA IS STORED
16915 ;          7816  !       AND ACCESSED FROM.
16916 ;          7817  !
16917 ;          7818  !       IO_BUF
16918 ;          7819  !
16919 ;          7820  !       A VECTOR OF 256 WORDS WHERE
16920 ;          7821  !       DATA FOR MBUS READS AND WRITE
16921 ;          7822  !       FUNCTION ARE FOUND.
16922 ;          7823  !
16923 ;          7824  !
16924 ;          7825  !--
16925 ;          7826
16926 ;          7827  local
16927 ;          7828      DODU_FLG,                   !DROP UNIT FLAG
16928 ;          7829      BG_PAT,                     !BACKGROUND PATTERN
16929 ;          7830      RESULT,                     !RESULTS FROM XOR
16930 ;          7831      TST_PAT;                    !TEST PATTERN
```

```
16932 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
16933 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (84)
16934
16935 ;          7832
16936 ;          7833  DODU_FLG = ZERO;
16937 ;          7834  BG_PAT = ZEROES;                       !BG PAT EQL 0'S
16938 ;          7835  TST_PAT = ONES;                        !TST PAT EQL 1'S
16939 ;          7836
16940 ;          7837  incr TWICE from 0 to 1 do              !REPEAT LOOP TWICE
16941 ;          7838      begin
16942 ;          7839      BGNSUB;
16943 ;          7840      CLR_MBUS;
16944 ;          7841      MLDT = .BG_PAT;                    !LOAD DATA DIAG REG WITH BG PAT
16945 ;          7842      MLD2 = .BG_PAT;
16946 ;          7843      MLE2 = .BG_PAT;
16947 ;          7844      DAT_DM_XFER ();                    !SET UP A DATA DIAG XFERR AT THE GOOD BLK
16948 ;          7845      MLCS1 = write;                     !DO A WRITE FUNCTION
16949 ;          7846
16950 ;          7847      incr WD_CNT from 0 to 127 do       !LOAD BLOCK WITH BG PAT
16951 ;          7848          begin
16952 ;          7849          DELAY (ONE_US);
16953 ;          7850          DAT_CLK = ONE;
16954 ;          7851          end;
16955 ;          7852
16956 ;          7853      CLR_MBUS;
16957 ;          7854      BAI = ONE;                         !SET ON FIRST IO_BUF ADRS
16958 ;          7855      IO_BUF = .TST_PAT;                 !FIRST IO_BUF WORD EQL'S TST_PAT
16959 ;          7856      GD_BLK_XFER ();                    !SET UP A GOOD BLOCK XFERR
16960 ;          7857      MLCS1 = write;                     !DO A WRITE FUNCTION
16961 ;          7858      TIME_OUT_LOOP;
16962 ;          7859      CLR_MBUS;
16963 ;          7860      DAT_DM_XFER ();                    !SET UP A DATA DIAG XFERR AT SAME BLOCK
16964 ;          7861      MLCS1 = read;                      !DO A READ FUNCTION
16965 ;          7862      DELAY (ONE_US);
16966 ;          7863
16967 ;          7864      incr WD_CNT from 0 to 112 do       !READ 113 LONG WORDS
16968 ;          7865          begin
16969 ;          7866          PD_TEMP = .MLPD;               !GET THE PROM DATA
16970 ;          7867          DAT_CLK = ONE;                 !CLOCK OUT THE DATA WORD
16971 ;          7868          DELAY (ONE_US);
16972 ;          7869          RD_LNG_WRD;                    !READ THE DATA WORD
16973 ;          7870
16974 ;          7871          incr NIB_PTR from 0 to 8 do    !LOOK AT 9 NIBBLES
16975 ;          7872
16976 ;          7873              if .PD_TEMP [.NIB_PTR] IS_NOT_SET   !FIND GOOD NIBBLES
16977 ;          7874              then
16978 ;          7875                  begin
16979 ;          7876                  XOR_LNG_WRD (.NIB_PTR, .TST_PAT, RESULT);       !XOR NIBBLE DATA WITH TST_PAT
16980 ;          7877
16981 ;          7878                  if .RESULT<0, 4> neq ZERO       !SEE IF EQUAL
16982 ;          7879                  then
16983 ;          7880                      begin                       !ERROR IF NEQ
16984 ;          7881                      ERRDF (86, SYNC, DUMPER);
16985 ;          7882                      PRINTB (FIV_FMT, WRD_24, WRD_25, WRD_10, WRD_12, FNC_5);
16986 ;          7883                      PRINTB (FMT_5, .TST_PAT, .RESULT, .NIB_PTR);
```

```
 16988 ;ML4                                                          14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
 16989 ;                                                             14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (84)
 16990
 16991 ;          7884                    DODU_FLG = UNE;
 16992 ;          7885                    end
 16993 ;          7886
 16994 ;          7887                end
 16995 ;          7888
 16996 ;          7889            end;
 16997 ;          7890
 16998 ;          7891        ENDSUB;
 16999 ;          7892
 17000 ;          7893        if .DODU_FLG IS_SET                    !DROP THIS UNIT IF DODU IS_SET
 17001 ;          7894                                               !AND EXITS TEST
 17002 ;          7895        then
 17003 ;          7896            begin
 17004 ;          7897            DODU (.ML_LUN);
 17005 ;          7898            DOCLN;
 17006 ;          7899            end;
 17007 ;          7900
 17008 ;          7901        BG_PAT = not .BG_PAT;                  !COMPLIMENT BG_PAT
 17009 ;          7902        TST_PAT = not .TST_PAT;                !COMPLIMENT TST_PAT AND REPEAT
 17010 ;          7903        end;
 17011 ;          7904
 17012 ;          7905    ENDTST;
 17016
 17017                                    .SBTTL  $T34
 17021 060530  004167  123362      $T34:  JSR     R1,$SAVE5                ;                              7778
 17022 060534  024646                     CMP     -(SP),-(SP)
 17023 060536  005046                     CLR     -(SP)                    ; DODU.FLG                     7833
 17024 060540  005001                     CLR     R1                       ; BG.PAT                       7834
 17025 060542  012702  177777             MOV     #-1,R2                   ; *,TST.PAT                    7835
 17026 060546  005046                     CLR     -(SP)                    ; TWICE                        7837
 17027 060550  104402                1$:  TRAP    2                        ;                              7838
 17028 060552  152777  000040  132434     BISB    #40,@ML.REG+40           ;                              7839
 17029 060560  016705  133016             MOV     ML.DUT,R5
 17030 060564  042705  177770             BIC     #177770,R5
 17031 060570  142777  000007  132416     BICB    #7,@ML.REG+40
 17032 060576  150577  132412             BISB    R5,@ML.REG+40
 17033 060602  010177  132536             MOV     R1,@ML.REG+170           ; BG.PAT,*                     7841
 17034 060606  010177  132542             MOV     R1,@ML.REG+200           ; BG.PAT,*                     7842
 17035 060612  010177  132516             MOV     R1,@ML.REG+160           ; BG.PAT,*                     7843
 17036 060616  004767  133340             JSR     PC,DAT.DM.XFER           ;                              7844
 17037 060622  012777  000061  132324     MOV     #61,@ML.REG              ;                              7845
 17038 060630  005003                     CLR     R3                       ; WD.CNT                       7847
 17039 060632  012704  000001          2$:MOV     #1,R4                    ; *,$$TMP2                     7849
 17040 060636  001411                3$:BEQ     6$
 17041 060640  016705  121252             MOV     L$DLY,R5                 ; *,$$TMP1
 17042 060644  001404                     BEQ     5$
```

```
                                                                                                14-May-1981 13:48:54    TOPS
                                                                                                14-May-1981 13:05:08    PA:<
17044                                   ;ML4
17045                                   ;
17046
17047 060646  005066  000006          4$:   CLR     6(SP)                   ; $$TMP
17048 060652  005305                        DEC     R5                      ; $$TMP1
17049 060654  001374                        BNE     4$
17050 060656  005304                  5$:   DEC     R4                      ; $$TMP2
17051 060660  000766                        BR      3$
17052 060662  152777  000020  132404  6$:   BISB    #20,@ML.REG+120         ;                                       7850
17053 060670  005203                        INC     R3                      ; WD.CNT                                7847
17054 060672  020327  000177                CMP     R3,#177                 ; WD.CNT,*
17055 060676  003755                        BLE     2$
17056 060700  152777  000040  132306        BISB    #40,@ML.REG+40          ;                                       7851
17057 060706  016705  132670                MOV     ML.DUT,R5
17058 060712  042705  177770                BIC     #177770,R5
17059 060716  142777  000007  132270        BICB    #7,@ML.REG+40
17060 060724  150577  132264                BISB    R5,@ML.REG+40
17061 060730  152777  000010  132256        BISB    #10,@ML.REG+40          ;                                       7854
17062 060736  010267  130636                MOV     R2,IO.BUF               ; TST.PAT,*                             7855
17063 060742  004767  135122                JSR     PC,GD.BLK.XFER          ;                                       7856
17064 060746  012777  000061  132200        MOV     #61,@ML.REG             ;                                       7857
17065 060754  105777  132244          7$:   TSTB    @ML.REG+50
17066 060760  100375                        BPL     7$
17067 060762  152777  000040  132224        BISB    #40,@ML.REG+40          ;                                       7858
17068 060770  016705  132606                MOV     ML.DUT,R5
17069 060774  042705  177770                BIC     #177770,R5
17070 061000  142777  000007  132206        BICB    #7,@ML.REG+40
17071 061006  150577  132202                BISB    R5,@ML.REG+40
17072 061012  004767  133144                JSR     PC,DAT.DM.XFER          ;                                       7860
17073 061016  012777  000071  132130        MOV     #71,@ML.REG             ;                                       7861
17074 061024  012704  000001                MOV     #1,R4                   ; *,$$TMP2                              7862
17075 061030  001411                  8$:   BEQ     11$
17076 061032  016705  121060                MOV     L$DLY,R5                ; *,$$TMP1
17077 061036  001404                        BEQ     10$
17078 061040  005066  000006          9$:   CLR     6(SP)                   ; $$TMP
17079 061044  005305                        DEC     R5                      ; $$TMP1
17080 061046  001374                        BNE     9$
17081 061050  005304                  10$:  DEC     R4                      ; $$TMP2
17082 061052  000766                        BR      8$
17083 061054  005003                  11$:  CLR     R3                      ; WD.CNT                                7864
17084 061056  017767  132322  132034  12$:  MOV     @ML.REG+230,PD.TEMP     ;                                       7866
17085 061064  152777  000020  132202        BISB    #20,@ML.REG+120         ;                                       7867
17086 061072  012704  000001                MOV     #1,R4                   ; *,$$TMP2                              7868
17087 061076  001411                  13$:  BEQ     16$
17088 061100  016705  121012                MOV     L$DLY,R5                ; *,$$TMP1
17089 061104  001404                        BEQ     15$
17090 061106  005066  000006          14$:  CLR     6(SP)                   ; $$TMP
17091 061112  005305                        DEC     R5                      ; $$TMP1
17092 061114  001374                        BNE     14$
17093 061116  005304                  15$:  DEC     R4                      ; $$TMP2
17094 061120  000766                        BR      13$
17095 061122  017767  132216  130022  16$:  MOV     @ML.REG+170,D1.TEMP
17096 061130  017767  132220  130016        MOV     @ML.REG+200,D2.TEMP
17097 061136  017767  132172  130012        MOV     @ML.REG+160,E2.TEMP
17098 061144  005004                        CLR     R4                      ; NIB.PTR                               7871
```

```
                                        ;ML4                                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                                       14-May-1981 13:05:08    PA:<
                                        17$:    MOV     R4,R5                   ; NIB.PTR,*                                             7873
                                                ASR     R5
                                                ASR     R5
                                                ASR     R5
                                                ADD     #PD.TEMP,R5
                                                MOV     R5,-(SP)
                                                MOV     R4,-(SP)                ; NIB.PTR,*
                                                BIC     #177770,(SP)
                                                MOV     #1,-(SP)
                                                CLR     -(SP)
                                                JSR     PC,BL$GT2
                                                ADD     #10,SP
                                                TST     R0
                                                BNE     19$
                                                MOV     R4,-(SP)                ; NIB.PTR,*                                             7876
                                                MOV     R2,-(SP)                ; TST.PAT,*
                                                MOV     #12,-(SP)
                                                ADD     SP,(SP)                 ; RESULT,*
                                                JSR     PC,XOR.LNG.WRD
                                                BIT     #17,12(SP)              ; *,RESULT                                              7878
                                                BEQ     18$
                                                TRAP    55                      ;                                                       7881
                                                .WORD   126
                                                .WORD   SYNC
                                                .WORD   DUMPER
                                                MOV     #FNC.5,-(SP)            ;                                                       7882
                                                MOV     #WRD.12,-(SP)
                                                MOV     #WRD.10,-(SP)
                                                MOV     #WRD.25,-(SP)
                                                MOV     #WRD.24,-(SP)
                                                MOV     #FIV.FMT,-(SP)
                                                MOV     #6,-(SP)
                                                MOV     SP,R0                   ; SP,*
                                                TRAP    14
                                                MOV     R4,(SP)                 ; NIB.PTR,*                                             7883
                                                MOV     30(SP),-(SP)            ; RESULT,*
                                                MOV     R2,-(SP)                ; TST.PAT,*
                                                MOV     #FMT.5,-(SP)
                                                MOV     #4,-(SP)
                                                MOV     SP,R0                   ; SP,*
                                                TRAP    14
                                                MOV     #1,36(SP)               ; *,DODU.FLG                                            7884
                                                ADD     #26,SP                  ;                                                       7890
                                        18$:    ADD     #6,SP                   ;                                                       7875
                                        19$:    INC     R4                      ; NIB.PTR                                               7871
                                                CMP     R4,#10                  ; NIB.PTR,*
                                                BLE     17$
                                                INC     R3                      ; WD.CNT                                                7864
                                                CMP     R3,#160                 ; WD.CNT,*
                                                BLE     12$
                                                TRAP    67                      ;                                                       7889
                                                ROR     R0
```

```
                                                                                                14-May-1981 13:48:54    TOPS
                                                                                                14-May-1981 13:05:08    PA:<
                        ;ML4
                        ;
                                BHIS    21$
                20$:    JMP     1$
                21$:    CMP     2(SP),#1                        ; DODU.FLG,*                            7893
                        BNE     22$
                        MOV     ML.LUN,R0                       ;                                       7897
                        TRAP    51
                        TRAP    44
                22$:    COM     R1                              ; BG.PAT                                7901
                        COM     R2                              ; TST.PAT                               7902
                        INC     (SP)                            ; TWICE                                 7837
                        CMP     (SP),#1                         ; TWICE,*
                        BLE     20$
                        ADD     #10,SP                          ;                                       7778
                        RTS     PC

                ; Routine Size:  232 words
                ; Maximum stack depth per invocation:  24 words


                        .SBTTL  T34
                T34::
                1$:     JSR     PC,$T34                         ;                                       7903
                        TRAP    66
                        ROR     R0
                        BLO     1$
                        RTS     PC

                ; Routine Size:  6 words
                ; Maximum stack depth per invocation:  0 words

;               7906    !<BLF/PAGE>
```

```
17207 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
17208 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (85)
17209
17210 ;          7907  !
17211 ;          7908  BGNTST;
17212 ;          7909
17213 ;          7910  !++
17214 ;          7911  ! TEST NUMBER:  TST 35
17215 ;          7912  !
17216 ;          7913  ! TEST NAME:    SYNC DATA BUS CONIINUITY /READ PATH
17217 ;          7914  !
17218 ;          7915  ! TEST DESCRIPTION:
17219 ;          7916  !       TEST THE CONITNUITY OF THE SYNCHRONOUS MODULE READ
17220 ;          7917  !       DATA BUS BY:
17221 ;          7918  !
17222 ;          7919  !       1.  VIA MBUS WRITE FUNCTION WRITE ONES INTO THE GOOD BLOCK.
17223 ;          7920  !
17224 ;          7921  !       2.  VIA MBUS READ FUNCTION READ THE GOOD BLOCK FOR ONES.
17225 ;          7922  !
17226 ;          7923  !       3.  REPEAT WITH COMPLIMENT DATA PATTERN.
17227 ;          7924  !
17228 ;          7925  ! IMPLICIT INPUTS:
17229 ;          7926  !       IO_BUF
17230 ;          7927  !       A VECTOR OF 256 WORDS WHERE DATA FOR MBUS READS AND WRITE
17231 ;          7928  !       FUNCTION ARE FOUND.
17232 ;          7929  !--
17233 ;          7930
17234 ;          7931  local
17235 ;          7932      DODU_FLG,                               !DROP UNIT FLAG
17236 ;          7933      TST_PAT,                                !TEST PATTERN
17237 ;          7934      BG_PAT;                                 !BACKGROUND PATTERN
17238 ;          7935
17239 ;          7936  DODU_FLG = ZERO;
17240 ;          7937  TST_PAT = ONES;
17241 ;          7938  BG_PAT = ZEROES;
17242 ;          7939
17243 ;          7940  incr TWICE from 0 to 1 do                   !REPEAT LOOP TWICE
17244 ;          7941      begin
17245 ;          7942      CLR_MBUS;
17246 ;          7943      BAI = ONE;                              !SET ON FIRST IO_BUF ADRS
17247 ;          7944      IO_BUF = .TST_PAT;                      !FIRST IO_BUF ADRS GET TST PAT
17248 ;          7945      GD_BLK_XFER ();                         !SET UP A GOOD BLOCK XFERR
17249 ;          7946      MLCS1 = write;                          !DO A WRITE FUNCTION (WRITES THE TST_PAT)
17250 ;          7947      TIME_OUT_LOOP;
17251 ;          7948      BGNSOB;
17252 ;          7949      CLR_MBUS;
17253 ;          7950
17254 ;          7951      incr IO_CNT from 0 to 255 do            !LOAD IO_BUF WITH BG PAT
17255 ;          7952          IO_BUF [.IO_CNT] = .BG_PAT;
17256 ;          7953
17257 ;          7954      GD_BLK_XFER ();                         !SET UP A GOOD BLOCK XFERR
17258 ;          7955      MLCS1 = read;                           !DO A READ FUNCTION (READ THE TST PAT)
17259 ;          7956      TIME_OUT_LOOP;
17260 ;          7957
17261 ;          7958      incr IO_CNT from 0 to 255 do            !READ THE IO_BUF FOR TEST PATTERN
```

```
 17263 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
 17264 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (85)
 17265
 17266 ;          7959
 17267 ;          7960          if .IO_BUF [.IO_CNT] neq .TST_PAT       !SEE IF IO_BUF WORD EQLS TST PAT
 17268 ;          7961          then
 17269 ;          7962              begin                               !ERROR AND SET DODU_FLG IF NEQ
 17270 ;          7963              ERRDF (87, SYNC, DUMPER);
 17271 ;          7964              PRINTB (SEV_FMT, WRD_24, WRD_25, WRD_10, WRD_12, WRD_23, FNC_6, WRD_19);
 17272 ;          7965              PRINTB (FMT_2, .TST_PAT, .IO_BUF [.IO_CNT], ^.TST_PAT xor .IO_BUF [.IO_CNT]));
 17273 ;          7966              DODU_FLG = ONE;
 17274 ;          7967              end;
 17275 ;          7968
 17276 ;          7969      ENDSUB;
 17277 ;          7970
 17278 ;          7971      if .DODU_FLG IS_SET                     !DROP THIS UNIT IF DODU_FLG IS SET
 17279 ;          7972      then
 17280 ;          7973          begin
 17281 ;          7974          DODU (.ML_LUN);
 17282 ;          7975          DOCLN;
 17283 ;          7976          end;
 17284 ;          7977
 17285 ;          7978      TST_PAT = not .TST_PAT;                 !COMPLIMENT TST_PAT
 17286 ;          7979      BG_PAT = not .BG_PAT;                   !COMPLIMENT BG_PAT AND REPEAT
 17287 ;          7980      end;
 17288 ;          7981
 17289 ;          7982  ENDTST;
 17293
 17294                                        .SBTTL  $T35
 17298 061464  004167  122426          $T35: JSR     R1,$SAVE5                 ;                                  7905
 17299 061470  005046                        CLR     -(SP)                     ; DODU.FLG                         7936
 17300 061472  012701  177777                MOV     #-1,R1                    ; *,TST.PAT                        7937
 17301 061476  005004                        CLR     R4                        ; BG.PAT                           7938
 17302 061500  005005                        CLR     R5                        ; TWICE                            7940
 17303 061502  152777  000040  131504  1$:   BISB    #40,@ML.REG+40            ;                                  7941
 17304 061510  016703  132066                MOV     ML.DUT,R3
 17305 061514  042703  177770                BIC     #177770,R3
 17306 061520  142777  000007  131466        BICB    #7,@ML.REG+40
 17307 061526  150377  131462                BISB    R3,@ML.REG+40
 17308 061532  152777  000010  131454        BISB    #10,@ML.REG+40            ;                                  7943
 17309 061540  010167  130034                MOV     R1,IO.BUF                 ; TST.PAT,*                        7944
 17310 061544  004767  134320                JSR     PC,GD.BLK.XFER            ;                                  7945
 17311 061550  012777  000061  131376        MOV     #61,@ML.REG               ;                                  7946
 17312 061556  105777  131442          2$:   TSTB    @ML.REG+50
 17313 061562  100375                        BPL     2$
 17314 061564  104402                  3$:   TRAP    2                         ;                                  7947
 17315 061566  152777  000040  131420        BISB    #40,@ML.REG+40            ;                                  7948
 17316 061574  016703  132002                MOV     ML.DUT,R3
 17317 061600  042703  177770                BIC     #177770,R3
```

```
17319                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
17320                                   ;                                                       14-May-1981 13:05:08    PA:<
17321
17322 061604  142777  000007  131402            BICB    #7,@ML.REG+40
17323 061612  150377  131376                    BISB    R3,@ML.REG+40
17324 061616  005002                            CLR     R2                      ; IO.CNT                                7951
17325 061620  010203                    4$:     MOV     R2,R3                   ; IO.CNT,*                              7952
17326 061622  006303                            ASL     R3
17327 061624  010463  011600                    MOV     R4,IO.BUF(R3)           ; BG.PAT,*
17328 061630  005202                            INC     R2                      ; IO.CNT                                7951
17329 061632  020227  000377                    CMP     R2,#377                 ; IO.CNT,*
17330 061636  003770                            BLE     4$
17331 061640  004767  134224                    JSR     PC,GD.BLK.XFER          ;                                       7954
17332 061644  012777  000071  131302            MOV     #71,@ML.REG             ;                                       7955
17333 061652  105777  131346            5$:     TSTB    @ML.REG+50
17334 061656  100375                            BPL     5$
17335 061660  005002                            CLR     R2                      ; IO.CNT                                7958
17336 061662  010203                    6$:     MOV     R2,R3                   ; IO.CNT,*                              7960
17337 061664  006303                            ASL     R3
17338 061666  062703  011600                    ADD     #IO.BUF,R3
17339 061672  021301                            CMP     (R3),R1                 ; *,TST.PAT
17340 061674  001454                            BEQ     7$
17341 061676  104455                            TRAP    55                      ;                                       7963
17342 061700  000127                            .WORD   127
17343 061702  010526                            .WORD   SYNC
17344 061704  024060                            .WORD   DUMPER
17345 061706  012746  006506                    MOV     #WRD.19,-(SP)           ;                                       7964
17346 061712  012746  010004                    MOV     #FNC.6,-(SP)
17347 061716  012746  006544                    MOV     #WRD.23,-(SP)
17348 061722  012746  006426                    MOV     #WRD.12,-(SP)
17349 061726  012746  006406                    MOV     #WRD.10,-(SP)
17350 061732  012746  006560                    MOV     #WRD.25,-(SP)
17351 061736  012746  006552                    MOV     #WRD.24,-(SP)
17352 061742  012746  006120                    MOV     #SEV.FMT,-(SP)
17353 061746  012746  000010                    MOV     #10,-(SP)
17354 061752  010600                            MOV     SP,R0                   ; SP,*
17355 061754  104414                            TRAP    14
17356 061756  011316                            MOV     (R3),(SP)               ;                                       7965
17357 061760  010146                            MOV     R1,-(SP)                ; TST.PAT,*
17358 061762  046616  000002                    BIC     2(SP),(SP)
17359 061766  040166  000002                    BIC     R1,2(SP)                ; TST.PAT,*
17360 061772  052616                            BIS     (SP)+,(SP)
17361 061774  011346                            MOV     (R3),-(SP)
17362 061776  010146                            MOV     R1,-(SP)                ; TST.PAT,*
17363 062000  012746  004266                    MOV     #FMT.2,-(SP)
17364 062004  012746  000004                    MOV     #4,-(SP)
17365 062010  010600                            MOV     SP,R0                   ; SP,*
17366 062012  104414                            TRAP    14
17367 062014  012766  000001  000032            MOV     #1,32(SP)               ; *,DODU.FLG                            7966
17368 062022  062706  000032                    ADD     #32,SP                  ;                                       7962
17369 062026  005202                    7$:     INC     R2                      ; IO.CNT                                7958
17370 062030  020227  000377                    CMP     R2,#377                 ; IO.CNT,*
17371 062034  003712                            BLE     6$
17372 062036  104467                            TRAP    67                      ;                                       7967
17373 062040  006000                            ROR     R0
```

```
                                ;ML4                                                    14-May-1981 13:48:54    TOPS
                                ;                                                       14-May-1981 13:05:08    PA:<
                                
062042  103650                          BLO     3$
062044  021627  000001                  CMP     (SP),#1                 ; DODU.FLG,*                    7971
062050  001004                          BNE     8$
062052  016700  131522                  MOV     ML.LUN,R0               ;                               7974
062056  104451                          TRAP    51
062060  104444                          TRAP    44
062062  005101                  8$:     COM     R1                      ; TST.PAT                       7978
062064  005104                          COM     R4                      ; BG.PAT                        7979
062066  005205                          INC     R5                      ; TWICE                         7940
062070  020527  000001                  CMP     R5,#1                   ; TWICE,*
062074  003602                          BLE     1$
062076  005726                          TST     (SP)+                   ;                               7905
062100  000207                          RTS     PC
                                
                                ; Routine Size:  135 words
                                ; Maximum stack depth per invocation:  20 words
                                
                                
                                
                                        .SBTTL  T35
062102                          T35::
062102  004767  177356          1$:     JSR     PC,$T35                 ;                               7980
062106  104466                          TRAP    66
062110  006000                          ROR     R0
062112  103773                          BLO     1$
062114  000207                          RTS     PC
                                
                                ; Routine Size:  6 words
                                ; Maximum stack depth per invocation:  0 words
                                
                                
                                ;       7983    !<BLF/PAGE>
```

```
17425 ;ML4
17426 :
17427
17428 ;          7984  !
17429 ;          7985  !
17430 ;          7986  BGNTST;
17431 ;          7987
17432 ;          7988  !++
17433 ;          7989  ! TEST NUMBER:  TST 36
17434 ;          7990  !
17435 ;          7991  ! TEST NAME:    RAM BUS ADRS COUNTER TEST /WRITE PATH
17436 ;          7992  !
17437 ;          7993  ! TEST DESCRIPTION:
17438 ;          7994  !
17439 ;          7995  !       TEST ABILITY OF THE RAM_BUS ADDRESS
17440 ;          7996  !       COUNTERS TO LOAD/UNLOAD THE SKIP
17441 ;          7997  !       RAM DURING WRITE FUNCTIONS BY:
17442 ;          7998  !
17443 ;          7999  !       1.  LOADING A REPEATING COUNT OF 0
17444 ;          8000  !           TO 63 INTO THE NIBBLES OF THE
17445 ;          8001  !           FIRST 64 WORDS OF THE IO_BUF.
17446 ;          8002  !
17447 ;          8003  !       2.  VIA MBUS WRITE FUNCTION WRITE
17448 ;          8004  !           THE CONTENTS OF THE IO_BUF
17449 ;          8005  !           INTO THE GOOD BLOCK.
17450 ;          8006  !
17451 ;          8007  !       3.  VIA DAT_DM READ GOOD NIBBLES IN
17452 ;          8008  !           THE GOOD BLOCK FOR THE UNBROKEN
17453 ;          8009  !           COUNT OF 0 TO 63.
17454 ;          8010  !
17455 ;          8011  !               ONCE A BAD NIBBLE IS ENCOUNTERED
17456 ;          8012  !               MASK THAT NIBBLE FROM FURTHER
17457 ;          8013  !               READS.
17458 ;          8014  !
17459 ;          8015  !       4.  REPEAT READING NIBBLES UNTIL
17460 ;          8016  !           113 WORDS ARE READ OR ALL 10 NIBBLES
17461 ;          8017  !           ARE MASKED.
17462 ;          8018  !
17463 ;          8019  ! IMPLICIT INPUTS:
17464 ;          8020  !
17465 ;          8021  !       PD_TEMP
17466 ;          8022  !
17467 ;          8023  !       A BIT VECTOR OF 16 BITS WHERE
17468 ;          8024  !       THE READ PROM DATA IS STORED
17469 ;          8025  !       AND ACCESSED FROM.
17470 ;          8026  !
17471 ;          8027  !       IO_BUF
17472 ;          8028  !       A VECTOR OF 256 WORDS WHERE
17473 ;          8029  !       DATA FOR MBUS READS AND WRITE
17474 ;          8030  !       FUNCTION ARE FOUND.
17475 ;          8031  !
17476 ;          8032  !--
17477 ;          8033
17478 ;          8034  local
17479 ;          8035      DODU_FLG,                                   !DROP UNIT FLAG
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (86)

;          8036      WRD_CNT,                           !WORD COUNT
;          8037      NIB_IGNORE : bitvector [16],       !NIBBLE IGNORE FLAGS
;          8038      ERR_FLG,                           !ERROR FLAG
;          8039      BAD_NIB_CNT,                       !BAD NIBBLE COUNT
;          8040      PASS_CNT,                          !PASS COUNT
;          8041      NIB_PAT;                           !NIBBLE PATTERN
;          8042
;          8043  DODU_FLG = ZERO;
;          8044  WRD_CNT = ZERO;
;          8045
;          8046  incr CNT from 0 to 63 do               !LOAD 64 WORDS WITH REPEATING COUNTS OF 0-63
;          8047
;          8048      incr PAT_CNT from -1 to 11 by 4 do  !LOAD NIBBLES IN WORD WITH REPEATING COUNTS OF 0-63
;          8049          begin
;          8050          (IO_BUF [.WRD_CNT])<0, 4> = .PAT_CNT + 1;      !LOAD FIRST NIBBLE IN WORD
;          8051          (IO_BUF [.WRD_CNT])<4, 4> = .PAT_CNT + 2;      !LOAD SECOND NIBBLE IN WORD
;          8052          (IO_BUF [.WRD_CNT])<8, 4> = .PAT_CNT + 3;      !LOAD THIRD NIBBLE IN WORD
;          8053          (IO_BUF [.WRD_CNT])<12, 4> = .PAT_CNT + 4;     !LOAD FORTH NIBBLE IN WORD
;          8054          WRD_CNT = .WRD_CNT + 1;            !INCREMENT TO NEXT WORD
;          8055          end;
;          8056
;          8057  BGNSUB;
;          8058  CLR_MBUS;
;          8059  GD_BLK_XFER ();                          !SET UP A GOOD BLOCK XFERR
;          8060  MLCS1 = write;                           !DO A WRITE FUNCTION
;          8061  TIME_OUT_LOOP;
;          8062  CLR_MBUS;
;          8063  NIB_IGNORE = ZEROES;
;          8064  PASS_CNT = ZEROES;
;          8065  NIB_PAT = ZEROES;
;          8066  BAD_NIB_CNT = ZEROES;
;          8067  DAT_DM_XFER ();                          !SET UP A DATA DIAG MODE AT THE GOOD BLOCK
;          8068  MLCS1 = read;                            !DO A READ FUNCTION
;          8069  DELAY (ONE_US);
;          8070
;          8071  do                                      !LOOP UNTIL THE BLOCK IS READ OR 9 BAD NIBBLES FOUND
;          8072      begin
;          8073      PD_TEMP = .MLPD;                    !GET THE PROM DATA
;          8074      DAT_CLK = ONE;                      !CLOCK OUT THE DATA WORD
;          8075      DELAY (ONE_US);
;          8076      RD_LNG_WRD;                         !READ THE DATA DIAG REGISTERS
;          8077
;          8078      incr NIB_PTR from 0 to 8 do         !LOOK AT 9 NIBBLES
;          8079          begin
;          8080
;          8081          if .PD_TEMP [.NIB_PTR] IS_NOT_SET       !FIND GOOD NIBBLES
;          8082          then
;          8083              begin
;          8084
;          8085              if .NIB_IGNORE [.NIB_PTR] IS_NOT_SET        !SEE IF THIS NIBBLE FOUND BAD BEFORE
;          8086              then
;          8087                  begin
```

```
;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
;                                                       14-May-1981 13:05:08    PA:<REALE>ML4.BLI.7 (86)

;          8088                 TST_LNG_WRD (.NIB_PTR, .NIB_PAT, ERR_FLG);      !TEST THE NIBBLE IF NEVER FOUND BAD
;          8089
;          8090                 if .ERR_FLG IS_SET                !SEE IF TEST FOUND AN ERROR
;          8091                 then
;          8092                     begin                         !ERROR AND SET DODU_FLG IS SET
;          8093                     ERRDF (88, ARR_DAT, DUMPER);
;          8094                     PRINTB (SIX_FMT, FNC_18, WRD_50, WRD_10, WRD_12, FNC_5, WRD_19);
;          8095                     DODU_FLG = ONE;
;          8096                     end
;          8097
;          8098                 end
;          8099
;          8100             end
;          8101             else                                  !THIS NIBBLE IS BAD
;          8102                 begin
;          8103                 NIB_IGNORE [.NIB_PTR] = ONE;      !SET THIS NIBBLE NIB_IGNORE FLAG
;          8104                 BAD_NIB_CNT = .BAD_NIB_CNT + 1;   !INCREMENT BAD NIB COUNT
;          8105                 end;
;          8106
;          8107             NIB_PAT = .NIB_PAT + 1;               !INCREMENT NIB PAT
;          8108             end;
;          8109
;          8110         PASS_CNT = .PASS_CNT + 1;                 !INCREMENT PASS COUNT
;          8111         end
;          8112     until (.PASS_CNT eql 113) or (.BAD_NIB_CNT eql 9);      !REPEAT UNTIL COMPLETE
;          8113
;          8114     ENDSUB;
;          8115
;          8116     if .DODU_FLG IS_SET                           !DROP THIS UNIT IF DODU_FLG SET
;          8117     then
;          8118         begin
;          8119         DODU (.ML_LUN);
;          8120         DOCLN;
;          8121         end;
;          8122
;          8123     ENDTST;
;
                                        .SBTTL  $T36
062116  004167  121774          $T36:   JSR     R1,$SAVE5                       ;                                       7982
062122  162706  000006                  SUB     #6,SP
062126  005046                          CLR     -(SP)                           ; DODU.FLG                              8043
062130  005003                          CLR     R3                              ; WRD.CNT                               8044
062132  005002                          CLR     R2                              ; CNT                                   8046
062134  012700  177777          1$:     MOV     #-1,R0                          ; *,PAT.CNT                             8048
062140  010301                  2$:     MOV     R3,R1                           ; WRD.CNT,*                             8050
062142  006301                          ASL     R1
```

```
17593                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
17594                                   ;                                                       14-May-1981 13:05:08    PA:<
17595
17596 062144  062701  011600                    ADD     #IO.BUF,R1
17597 062150  010005                            MOV     R0,R5                   ; PAT.CNT,*
17598 062152  005205                            INC     R5
17599 062154  042705  177760                    BIC     #177760,R5
17600 062160  142711  000017                    BICB    #17,(R1)
17601 062164  150511                            BISB    R5,(R1)
17602 062166  010005                            MOV     R0,R5                   ; PAT.CNT,*             8051
17603 062170  062705  000002                    ADD     #2,R5
17604 062174  006305                            ASL     R5
17605 062176  006305                            ASL     R5
17606 062200  006305                            ASL     R5
17607 062202  006305                            ASL     R5
17608 062204  042705  177417                    BIC     #177417,R5
17609 062210  142711  000360                    BICB    #360,(R1)
17610 062214  150511                            BISB    R5,(R1)
17611 062216  010005                            MOV     R0,R5                   ; PAT.CNT,*             8052
17612 062220  062705  000003                    ADD     #3,R5
17613 062224  000305                            SWAB    R5
17614 062226  042705  170377                    BIC     #170377,R5
17615 062232  042711  007400                    BIC     #7400,(R1)
17616 062236  050511                            BIS     R5,(R1)
17617 062240  010005                            MOV     R0,R5                   ; PAT.CNT,*             8053
17618 062242  062705  000004                    ADD     #4,R5
17619 062246  000305                            SWAB    R5
17620 062250  006305                            ASL     R5
17621 062252  006305                            ASL     R5
17622 062254  006305                            ASL     R5
17623 062256  006305                            ASL     R5
17624 062260  042705  007777                    BIC     #7777,R5
17625 062264  042711  170000                    BIC     #170000,(R1)
17626 062270  050511                            BIS     R5,(R1)
17627 062272  005203                            INC     R3                      ; WRD.CNT               8054
17628 062274  062700  000004                    ADD     #4,R0                   ; *,PAT.CNT             8048
17629 062300  020027  000013                    CMP     R0,#13                  ; PAT.CNT,*
17630 062304  003715                            BLE     2$
17631 062306  005202                            INC     R2                      ; CNT                   8046
17632 062310  020227  000077                    CMP     R2,#77                  ; CNT,*
17633 062314  003707                            BLE     1$
17634 062316  104402                    3$:     TRAP    2                       ;                       8055
17635 062320  152777  000040  130666            BISB    #40,@ML.REG+40          ;                       8057
17636 062326  016702  131250                    MOV     ML.DUT,R2
17637 062332  042702  177770                    BIC     #177770,R2
17638 062336  142777  000007  130650            BICB    #7,@ML.REG+40
17639 062344  150277  130644                    BISB    R2,@ML.REG+40
17640 062350  004767  133514                    JSR     PC,GD.BLK.XFER          ;                       8059
17641 062354  012777  000061  130572            MOV     #61,@ML.REG             ;                       8060
17642 062362  105777  130636            4$:     TSTB    @ML.REG+50
17643 062366  100375                            BPL     4$
17644 062370  152777  000040  130616            BISB    #40,@ML.REG+40          ;                       8061
17645 062376  016702  131200                    MOV     ML.DUT,R2
17646 062402  042702  177770                    BIC     #177770,R2
17647 062406  142777  000007  130600            BICB    #7,@ML.REG+40
```

```
17649                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
17650                                   ;                                                       14-May-1981 13:05:08    PA:<
17651
17652 062414  150277  130574                    BISB    R2,@ML.REG+40
17653 062420  005066  000004                    CLR     4(SP)           ; NIB.IGNORE                            8063
17654 062424  005004                            CLR     R4              ; PASS.CNT                              8064
17655 062426  005003                            CLR     R3              ; NIB.PAT                               8065
17656 062430  005005                            CLR     R5              ; BAD.NIB.CNT                           8066
17657 062432  004767  131524                    JSR     PC,DAT.DM.XFER  ;                                       8067
17658 062436  012777  000071  130510            MOV     #71,@ML.REG     ;                                       8068
17659 062444  012701  000001                    MOV     #1,R1           ; *,$$TMP2                              8069
17660 062450  001411                    5$:     BEQ     8$
17661 062452  016702  117440                    MOV     L$DLY,R2        ; *,$$TMP1
17662 062456  001404                            BEQ     7$
17663 062460  005066  000006            6$:     CLR     6(SP)           ; $$TMP
17664 062464  005302                            DEC     R2              ; $$TMP1
17665 062466  001374                            BNE     6$
17666 062470  005301                    7$:     DEC     R1              ; $$TMP2
17667 062472  000766                            BR      5$
17668 062474  017767  130704  130416    8$:     MOV     @ML.REG+230,PD.TEMP     ;                       8073
17669 062502  152777  000020  130564            BISB    #20,@ML.REG+120 ;                                       8074
17670 062510  012701  000001                    MOV     #1,R1           ; *,$$TMP2                              8075
17671 062514  001411                    9$:     BEQ     12$
17672 062516  016702  117374                    MOV     L$DLY,R2        ; *,$$TMP1
17673 062522  001404                            BEQ     11$
17674 062524  005066  000006            10$:    CLR     6(SP)           ; $$TMP
17675 062530  005302                            DEC     R2              ; $$TMP1
17676 062532  001374                            BNE     10$
17677 062534  005301                    11$:    DEC     R1              ; $$TMP2
17678 062536  000766                            BR      9$
17679 062540  017767  130600  126404    12$:    MOV     @ML.REG+170,D1.TEMP
17680 062546  017767  130602  126400            MOV     @ML.REG+200,D2.TEMP
17681 062554  017767  130554  126374            MOV     @ML.REG+160,E2.TEMP
17682 062562  005001                            CLR     R1              ; NIB.PTR                               8078
17683 062564  010100                    13$:    MOV     R1,R0           ; NIB.PTR,*                             8081
17684 062566  006200                            ASR     R0
17685 062570  006200                            ASR     R0
17686 062572  006200                            ASR     R0
17687 062574  012702  000004                    MOV     #4,R2           ;                                       8085
17688 062600  060602                            ADD     SP,R2           ; NIB.IGNORE,*
17689 062602  060002                            ADD     R0,R2
17690 062604  010046                            MOV     R0,-(SP)        ;                                       8081
17691 062606  062716  013120                    ADD     #PD.TEMP,(SP)
17692 062612  010146                            MOV     R1,-(SP)        ; NIB.PTR,*
17693 062614  042716  177770                    BIC     #177770,(SP)
17694 062620  012746  000001                    MOV     #1,-(SP)
17695 062624  005046                            CLR     -(SP)
17696 062626  004767  120306                    JSR     PC,BL$GT2
17697 062632  062706  000010                    ADD     #10,SP
17698 062636  005700                            TST     R0
17699 062640  001066                            BNE     15$
17700 062642  010246                            MOV     R2,-(SP)        ;                                       8085
17701 062644  010146                            MOV     R1,-(SP)        ; NIB.PTR,*
17702 062646  042716  177770                    BIC     #177770,(SP)
17703 062652  012746  000001                    MOV     #1,-(SP)
```

```
17705                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
17706                                   ;                                                       14-May-1981 13:05:08    PA:<
17707
17708 062656  005046                            CLR     -(SP)
17709 062660  004767  120254                    JSR     PC,BL$GT2
17710 062664  062706  000010                    ADD     #10,SP
17711 062670  005700                            TST     R0
17712 062672  001065                            BNE     16$
17713 062674  010146                            MOV     R1,-(SP)        ; NIB.PTR,*                             8088
17714 062676  010346                            MOV     R3,-(SP)        ; NIB.PAT,*
17715 062700  012746  000010                    MOV     #10,-(SP)
17716 062704  060616                            ADD     SP,(SP)         ; ERR.FLG,*
17717 062706  004767  133242                    JSR     PC,TST.LNG.WRD
17718 062712  026627  000010  000001            CMP     10(SP),#1       ; ERR.FLG,*                             8090
17719 062720  001033                            BNE     14$
17720 062722  104455                            TRAP    55              ;                                       8093
17721 062724  000130                            .WORD   130
17722 062726  010570                            .WORD   ARR.DAT
17723 062730  024060                            .WORD   DUMPER
17724 062732  012746  006506                    MOV     #WRD.19,-(SP)   ;                                       8094
17725 062736  012746  007774                    MOV     #FNC.5,-(SP)
17726 062742  012746  006426                    MOV     #WRD.12,-(SP)
17727 062746  012746  006406                    MOV     #WRD.10,-(SP)
17728 062752  012746  007046                    MOV     #WRD.50,-(SP)
17729 062756  012746  010172                    MOV     #FNC.18,-(SP)
17730 062762  012746  006100                    MOV     #SIX.FMT,-(SP)
17731 062766  012746  000007                    MOV     #7,-(SP)
17732 062772  010600                            MOV     SP,R0           ; SP,*
17733 062774  104414                            TRAP    14
17734 062776  012766  000001  000026            MOV     #1,26(SP)       ; *,DODU.FLG                            8095
17735 063004  062706  000020                    ADD     #20,SP          ;                                       8092
17736 063010  062706  000006          14$:      ADD     #6,SP           ;                                       8087
17737 063014  000414                            BR      16$             ;                                       8081
17738 063016  010246          15$:      MOV     R2,-(SP)        ;                                       8103
17739 063020  010146                            MOV     R1,-(SP)        ; NIB.PTR,*
17740 063022  042716  177770                    BIC     #177770,(SP)
17741 063026  012746  000001                    MOV     #1,-(SP)
17742 063032  011646                            MOV     (SP),-(SP)
17743 063034  004767  120336                    JSR     PC,BL$PU2
17744 063040  005205                            INC     R5              ; BAD.NIB.CNT                           8104
17745 063042  062706  000010                    ADD     #10,SP          ;                                       8102
17746 063046  005203          16$:      INC     R3              ; NIB.PAT                               8107
17747 063050  005201                            INC     R1              ; NIB.PTR                               8078
17748 063052  020127  000010                    CMP     R1,#10          ; NIB.PTR,*
17749 063056  003642                            BLE     13$
17750 063060  005204                            INC     R4              ; PASS.CNT                              8110
17751 063062  020427  000161                    CMP     R4,#161         ; PASS.CNT,*                            8112
17752 063066  001405                            BEQ     17$
17753 063070  020527  000011                    CMP     R5,#11          ; BAD.NIB.CNT,*
17754 063074  001402                            BEQ     17$
17755 063076  000167  177372                    JMP     8$
17756 063102  104467          17$:      TRAP    67
17757 063104  006000                            ROR     R0
17758 063106  103002                            BHIS    18$
17759 063110  000167  177202                    JMP     3$
```

```
                                ;ML4                                                    14-May-1981 13:48:54    TOPS
                                ;                                                       14-May-1981 13:05:08    PA:<
                                18$:    CMP     (SP),#1                 ; DODU.FLG,*                            8116
                                        BNE     19$
                                        MOV     ML.LUN,R0               ;                                       8119
                                        TRAP    51
                                        TRAP    44
                                19$:    ADD     #10,SP                  ;                                       7982
                                        RTS     PC

                                ; Routine Size:  265 words
                                ; Maximum stack depth per invocation:  21 words


                                        .SBTTL  T36
                                T36::
                                1$:     JSR     PC,$T36                 ;                                       8121
                                        TRAP    56
                                        ROR     R0
                                        BLO     1$
                                        RTS     PC

                                ; Routine Size:  6 words
                                ; Maximum stack depth per invocation:  0 words


;               8124    !<BLF/PAGE>
```

```
17805 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
17806 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (87)
17807
17808 ;          8125  !
17809 ;          8126  !
17810 ;          8127  BGNTST;
17811 ;          8128
17812 ;          8129  !++
17813 ;          8130  ! TEST NUMBER:  TST 37
17814 ;          8131  !
17815 ;          8132  ! TEST NAME:    RAM BUS ADRS COUNTER TEST /READ PATH
17816 ;          8133  !
17817 ;          8134  ! TEST DESCRIPTION:
17818 ;          8135  !
17819 ;          8136  !       TEST ABILITY OF RAM/BUS ADRS
17820 ;          8137  !       COUNTERS TO LOAD/UNLOAD THE SKIP
17821 ;          8138  !       RAM DURING READ FUNCTIONS BY:
17822 ;          8139  !
17823 ;          8140  !       1.  LOADING A REPEATING COUNT OF 0
17824 ;          8141  !           TO 63 INTO THE NIBBLES OF THE
17825 ;          8142  !           FIRST 64 WORDS OF THE IO_BUF.
17826 ;          8143  !
17827 ;          8144  !       2.  VIA MBUS WRITE FUNCTION WRITE
17828 ;          8145  !           THE CONTENTS OF THE IO_BUF
17829 ;          8146  !           INTO THE GOOD BLOCK.
17830 ;          8147  !
17831 ;          8148  !       3.  CLEAR OUT THE IO_BUF
17832 ;          8149  !
17833 ;          8150  !       4.  VIA MBUS READ FUNCTION READ
17834 ;          8151  !           THE GOOD BLOCK FOR THE REPEATING
17835 ;          8152  !           COUNT OF 0 TO 63.
17836 ;          8153  !
17837 ;          8154  ! IMPLICIT INPUTS:
17838 ;          8155  !
17839 ;          8156  !       IO_BUF
17840 ;          8157  !
17841 ;          8158  !       A VECTOR OF 256 WORDS WHERE
17842 ;          8159  !       DATA FOR MBUS READS AND WRITE
17843 ;          8160  !       FUNCTION ARE FOUND.
17844 ;          8161  !
17845 ;          8162  !--
17846 ;          8163
17847 ;          8164  local
17848 ;          8165      DODU_FLG,                          !DROP UNIT FLAG
17849 ;          8166      WRD_CNT,                           !WORD COUNT
17850 ;          8167      PAT_INC,                           !PATTERN INCREMENT
17851 ;          8168      SIZ_EXP,                           !SIZE EXPRESSION
17852 ;          8169      POS_EXP,                           !POSITIONAL EXPRESSION
17853 ;          8170      TEMP;                              !TEMPORARY STORAGE LOCATION
17854 ;          8171
17855 ;          8172  CLR_MBUS;
17856 ;          8173  DODU_FLG = ZERO;
17857 ;          8174  WRD_CNT = ZERO;
17858 ;          8175
17859 ;          8176  incr COUNT from 0 to 63 do             !LOAD 64 WORDS WITH REPEATING  COUNTS OF 0-63
```

```
17861 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
17862 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (87)
17863 ;
17864 ;          8177
17865 ;          8178      incr PAT_CNT from -1 to 11 by 4 do          !LOAD NIBBLES IN WORD WITH REPEATING COUNT OF 0-63
17866 ;          8179          begin
17867 ;          8180          (IO_BUF [.WRD_CNT])<0, 4> = .PAT_CNT + 1;       !LOAD FIRST NIBBLE IN WORD
17868 ;          8181          (IO_BUF [.WRD_CNT])<4, 4> = .PAT_CNT + 2;       !LOAD SECOND NIBBLE IN WORD
17869 ;          8182          (IO_BUF [.WRD_CNT])<8, 4> = .PAT_CNT + 3;       !LOAD THIRD NIBBLE IN WORD
17870 ;          8183          (IO_BUF [.WRD_CNT])<12, 4> = .PAT_CNT + 4;      !LOAD FORTH NIBBLE IN WORD
17871 ;          8184          WRD_CNT = .WRD_CNT + 1;                         !INCREMENT TO NEXT WORD
17872 ;          8185          end;
17873 ;          8186
17874 ;          8187  GD_BLK_XFER ();                                        !SET UP A GOOD BLOCK XFERR
17875 ;          8188  ML[S1 = write;                                         !DO A WRITE FUNCTION
17876 ;          8189  TIME_OUT_LOOP;
17877 ;          8190  BGNSOB;
17878 ;          8191
17879 ;          8192  incr IO_CNT from 0 to 255 do                          !CLEAR OUT IO_BUF
17880 ;          8193      IO_BUF [.IO_CNT] = ZEROES;
17881 ;          8194
17882 ;          8195  CLR_MBUS;
17883 ;          8196  GD_BLK_XFER ();                                        !SET UP A GOOD BLOCK XFERR
17884 ;          8197  ML[S1 = read;                                          !DO A READ FUNCTION
17885 ;          8198  TIME_OUT_LOOP;
17886 ;          8199  CLR_MBUS;
17887 ;          8200  SIZ_EXP = 4;                                           !FIELD SIZE FOR NIBBLES ALWAYS 4 BITS
17888 ;          8201  WRD_CNT = 0;
17889 ;          8202
17890 ;          8203  incr COUNT from 0 to 63 do                             !READ 64 WORDS IN IO_BUF
17891 ;          8204
17892 ;          8205      incr PAT_CNT from -1 to 11 by 4 do                 !READ REPEATING COUNTS OF 0-63
17893 ;          8206          begin
17894 ;          8207          POS_EXP = ZERO;                                !FIELD SELECTOR SELECTS THE FOUR NIBBLES
17895 ;          8208          PAT_INC = ONE;
17896 ;          8209          TEMP = .IO_BUF [.WRD_CNT];                     !GET A WORD OUT OF IO_BUF
17897 ;          8210
17898 ;          8211          incr CNT from 0 to 3 do                        !READ THE FOUR NIBBLES IN WORD
17899 ;          8212              begin
17900 ;          8213
17901 ;          8214              if .TEMP<.POS_EXP, .SIZ_EXP> neq (.PAT_CNT + .PAT_INC)
17902 ;          8215                                                         !COMPARE NIBBLE WITH RESPECTIVE 0 -63 CNT
17903 ;          8216              then
17904 ;          8217                  begin                                !ERROR AND SET DODU_FLG IF NEQ
17905 ;          8218                  ERRDF (89, ARR_DAT, DUMPER);
17906 ;          8219                  PRINTB (SIX_FMT, FNC_18, WRD_50, WRD_10, WRD_12, FNC_6, WRD_19);
17907 ;          8220                  DODU_FLG = ONE;
17908 ;          8221                  end;
17909 ;          8222
17910 ;          8223              POS_EXP = .POS_EXP + 4;                    !POINT TO THE NEXT NIBBLE IN WORD
17911 ;          8224              PAT_INC = .PAT_INC + 1;                    !INCREMENT THE 0-63 COUNT
17912 ;          8225              end;
17913 ;          8226
17914 ;          8227          WRD_CNT = .WRD_CNT + 1;                        !GET THE NEXT IO_BUF WORD
17915 ;          8228          end;
```

```
17917 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
17918 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (87)
17919
17920 ;          8229
17921 ;          8230  ENDSUB;
17922 ;          8231
17923 ;          8232  if .DODU_FLG IS_SET                    !DROP THIS UNIT IF DODU_FLG SET
17924 ;          8233  then
17925 ;          8234      begin
17926 ;          8235      DODU (.ML_LUN);
17927 ;          8236      DOCLN;
17928 ;          8237      end;
17929 ;          8238
17930 ;          8239  ENDTST;
17934
17935                                          .SBTTL  $T37
17939 063154  004167  120736          $T37:   JSR     R1,$SAVE5                       ;                       8123
17940 063160  162706  000012                  SUB     #12,SP
17941 063164  152777  000040  130022          BISB    #40,@ML.REG+40                  ;                       8170
17942 063172  016705  130404                  MOV     ML.DUT,R5
17943 063176  042705  177770                  BIC     #177770,R5
17944 063202  142777  000007  130004          BICB    #7,@ML.REG+40
17945 063210  150577  130000                  BISB    R5,@ML.REG+40
17946 063214  005066  000010                  CLR     10(SP)                          ; DODU.FLG              8173
17947 063220  005066  000002                  CLR     2(SP)                           ; WRD.CNT               8174
17948 063224  005002                          CLR     R2                              ; COUNT                 8176
17949 063226  012703  177777          1$:     MOV     #-1,R3                          ; *,PAT.CNT             8178
17950 063232  016604  000002          2$:     MOV     2(SP),R4                        ; WRD.CNT,*             8180
17951 063236  006304                          ASL     R4
17952 063240  062704  011600                  ADD     #IO.BUF,R4
17953 063244  010305                          MOV     R3,R5                           ; PAT.CNT,*
17954 063246  005205                          INC     R5
17955 063250  042705  177760                  BIC     #177760,R5
17956 063254  142714  000017                  BICB    #17,(R4)
17957 063260  150514                          BISB    R5,(R4)
17958 063262  010305                          MOV     R3,R5                           ; PAT.CNT,*             8181
17959 063264  062705  000002                  ADD     #2,R5
17960 063270  006305                          ASL     R5
17961 063272  006305                          ASL     R5
17962 063274  006305                          ASL     R5
17963 063276  006305                          ASL     R5
17964 063300  042705  177417                  BIC     #177417,R5
17965 063304  142714  000360                  BICB    #360,(R4)
17966 063310  150514                          BISB    R5,(R4)
17967 063312  010305                          MOV     R3,R5                           ; PAT.CNT,*             8182
17968 063314  062705  000003                  ADD     #3,R5
17969 063320  000305                          SWAB    R5
17970 063322  042705  170377                  BIC     #170377,R5
17971 063326  042714  007400                  BIC     #7400,(R4)
```

```
                                                                                                14-May-1981 13:48:54    TOPS
                                ;ML4                                                            14-May-1981 13:05:08    PA:<
                                :
                                        BIS     R5,(R4)
                                        MOV     R3,R5                   ; PAT.CNT,*                                     8183
                                        ADD     #4,R5
                                        SWAB    R5
                                        ASL     R5
                                        ASL     R5
                                        ASL     R5
                                        ASL     R5
                                        BIC     #7777,R5
                                        BIC     #170000,(R4)
                                        BIS     R5,(R4)
                                        INC     2(SP)                   ; WRD.CNT                                       8184
                                        ADD     #4,R3                   ; *,PAT.CNT                                     8178
                                        CMP     R3,#13                  ; PAT.CNT,*
                                        BLE     2$
                                        INC     R2                      ; COUNT                                         8176
                                        CMP     R2,#77                  ; COUNT,*
                                        BLE     1$
                                        JSR     PC,GD.BLK.XFER          ;                                               8187
                                        MOV     #61,@ML.REG             ;                                               8188
                                3$:     TSTB    @ML.REG+50
                                        BPL     3$
                                4$:     TRAP    2                       ;                                               8189
                                        CLR     R3                      ; IO.CNT                                        8192
                                5$:     MOV     R3,R4                   ; IO.CNT,*                                      8193
                                        ASL     R4
                                        CLR     IO.BUF(R4)
                                        INC     R3                      ; IO.CNT                                        8192
                                        CMP     R3,#377                 ; IO.CNT,*
                                        BLE     5$
                                        BISB    #40,@ML.REG+40          ;                                               8193
                                        MOV     ML.DUT,R4
                                        BIC     #177770,R4
                                        BICB    #7,@ML.REG+40
                                        BISB    R4,@ML.REG+40
                                        JSR     PC,GD.BLK.XFER          ;                                               8196
                                        MOV     #71,@ML.REG             ;                                               8197
                                6$:     TSTB    @ML.REG+50
                                        BPL     6$
                                        BISB    #40,@ML.REG+40          ;                                               8198
                                        MOV     ML.DUT,R4
                                        BIC     #177770,R4
                                        BICB    #7,@ML.REG+40
                                        BISB    R4,@ML.REG+40
                                        MOV     #4,6(SP)                ; *,SIZ.EXP                                     8200
                                        CLR     2(SP)                   ; WRD.CNT                                       8201
                                        CLR     R1                      ; COUNT                                         8203
                                7$:     MOV     #-1,R3                  ; *,PAT.CNT                                     8205
                                8$:     CLR     R5                      ; POS.EXP                                       8207
                                        MOV     #1,(SP)                 ; *,PAT.INC                                     8208
                                        MOV     2(SP),R4                ; WRD.CNT,*                                     8209
                                        ASL     R4
```

```
                                                                                                14-May-1981 13:48:54    TOPS
                                                                                                14-May-1981 13:05:08    PA:<
18029                           ;ML4
18030                           ;
18031
18032 063614  016466  011600  000004          MOV     IO.BUF(R4),4(SP)        ; *,TEMP
18033 063622  005002                          CLR     R2                      ; CNT                                   8211
18034 063624  016646  000004          9$:     MOV     4(SP),-(SP)             ; TEMP,*                                8214
18035 063630  010546                          MOV     R5,-(SP)                ; POS.EXP,*
18036 063632  016646  000012                  MOV     12(SP),-(SP)            ; SIZ.EXP,*
18037 063636  005046                          CLR     -(SP)
18038 063640  004767  117152                  JSR     PC,BL$GT1
18039 063644  062706  000010                  ADD     #10,SP
18040 063650  010304                          MOV     R3,R4                   ; PAT.CNT,*
18041 063652  061604                          ADD     (SP),R4                 ; PAT.INC,*
18042 063654  020004                          CMP     R0,R4
18043 063656  001433                          BEQ     10$
18044 063660  104455                          TRAP    55                      ;                                       8218
18045 063662  000131                          .WORD   131
18046 063664  010570                          .WORD   ARR.DAT
18047 063666  024060                          .WORD   D MPER
18048 063670  012746  006506                  MOV     #WRD.19,-(SP)           ;                                       8219
18049 063674  012746  010004                  MOV     #FNC.6,-(SP)
18050 063700  012746  006426                  MOV     #WRD.12,-(SP)
18051 063704  012746  006406                  MOV     #WRD.10,-(SP)
18052 063710  012746  007046                  MOV     #WRD.50,-(SP)
18053 063714  012746  010172                  MOV     #FNC.18,-(SP)
18054 063720  012746  006100                  MOV     #SIX.FMT,-(SP)
18055 063724  012746  000007                  MOV     #7,-(SP)
18056 063730  010600                          MOV     SP,R0                   ; SP,*
18057 063732  104414                          TRAP    14
18058 063734  012766  000001  000030          MOV     #1,30(SP)               ; *,DODU.FLG                            8220
18059 063742  062706  000020                  ADD     #20,SP                  ;                                       8217
18060 063746  062705  000004          10$:    ADD     #4,R5                   ; *,POS.EXP                             8223
18061 063752  005216                          INC     (SP)                    ; PAT.INC                               8224
18062 063754  005202                          INC     R2                      ; CNT                                   8211
18063 063756  020227  000003                  CMP     R2,#3                   ; CNT,*
18064 063762  003720                          BLE     9$
18065 063764  005266  000002                  INC     2(SP)                   ; WRD.CNT                               8227
18066 063770  062703  000004                  ADD     #4,R3                   ; *,PAT.CNT                             8205
18067 063774  020327  000013                  CMP     R3,#13                  ; PAT.CNT,*
18068 064000  003677                          BLE     8$
18069 064002  005201                          INC     R1                      ; COUNT                                 8203
18070 064004  020127  000077                  CMP     R1,#77                  ; COUNT,*
18071 064010  003671                          BLE     7$
18072 064012  104467                          TRAP    67                      ;                                       8228
18073 064014  006000                          ROR     R0
18074 064016  103606                          BLO     4$
18075 064020  026627  000010  000001          CMP     10(SP),#1               ; DODU.FLG,*                            8232
18076 064026  001004                          BNE     11$
18077 064030  016700  127544                  MOV     ML.LUN,R0               ;                                       8235
18078 064034  104451                          TRAP    51
18079 064036  104444                          TRAP    44
18080 064040  062706  000012          11$:    ADD     #12,SP                  ;                                       8123
18081 064044  000207                          RTS     PC
18082
18083                           ; Routine Size:  221 words
```

```
 18085                                  ;ML4                                                    14-May-1981 13:48:54    TOPS
 18086                                  ;                                                       14-May-1981 13:05:08    PA:<
 18087
 18088                                  ; Maximum stack depth per invocation:  19 words
 18093
 18094
 18098
 18099                                          .SBTTL  T37
 18103 064046                           T37::
 18104 064046  004767  177102           1$:     JSR     PC,$T37                         ;       8237
 18105 064052  104466                           TRAP    66
 18106 064054  006000                           ROR     R0
 18107 064056  103773                           BLO     1$
 18108 064060  000207                           RTS     PC
 18109
 18110                                  ; Routine Size:  6 words
 18111                                  ; Maximum stack depth per invocation:  0 words
 18116
 18117
 18118 ;          8240  !<BLF/PAGE>
```

```
18120 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
18121 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (88)
18122
18123 ;          8241  !
18124 ;          8242  !
18125 ;          8243  BGNTST;
18126 ;          8244
18127 ;          8245  !++
18128 ;          8246  ! TEST NUMBER:  TST 38
18129 ;          8247  !
18130 ;          8248  ! TEST NAME:    SYNC DATA BUS BIT UNIQUENESS TEST/WRITE PATH
18131 ;          8249  !
18132 ;          8250  ! TEST DESCRIPTION:
18133 ;          8251  !
18134 ;          8252  !       TEST SYNCHRONOUS DATA BUS FOR
18135 ;          8253  !       DATA BIT UNIQUENESS BY:
18136 ;          8254  !
18137 ;          8255  !       1.  LOADING THE FIRST 16 WORDS IN
18138 ;          8256  !           THE IO_BUF WITH A SHIFTING
18139 ;          8257  !           ZERO IN A FIELD OF ONES PATTERN.
18140 ;          8258  !
18141 ;          8259  !       2.  VIA MBUS WRITE FUNCTION WRITE
18142 ;          8260  !           SHIFTING PATTERN THROUGH THE
18143 ;          8261  !           DATA BUS AND INTO THE GOOD
18144 ;          8262  !           BLOCK.
18145 ;          8263  !
18146 ;          8264  !       3.  VIA DAT_DM MODE READ THE
18147 ;          8265  !           GOOD BLOCK AND SAVE ALL GOOD
18148 ;          8266  !           NIBBLE DATA, IN THEIR PROPER
18149 ;          8267  !           SEQUENCE, INTO A STACK
18150 ;          8268  !           STRUCTURE.
18151 ;          8269  !
18152 ;          8270  !       4.  INTERRIGATE STACK STRUCTURE FOR
18153 ;          8271  !           SHIFTED DATA PATTERN.
18154 ;          8272  !
18155 ;          8273  !
18156 ;          8274  ! IMPLICIT INPUTS:
18157 ;          8275  !
18158 ;          8276  !       PD_TEMP
18159 ;          8277  !       A BIT VECTOR OF 16 BITS WHERE
18160 ;          8278  !       THE READ PROM DATA IS STORED
18161 ;          8279  !       AND ACCESSED FROM.
18162 ;          8280  !
18163 ;          8281  !       IO_BUF
18164 ;          8282  !       A VECTOR OF 256 WORDS WHERE
18165 ;          8283  !       DATA FOR MBUS READS AND WRITE
18166 ;          8284  !       FUNCTION ARE FOUND.
18167 ;          8285  !
18168 ;          8286  !       STACK
18169 ;          8287  !       A VECTOR OF 198 BYTE LOCATIONS
18170 ;          8288  !       WHERE GOOD NIBBLE DATA IS STORED
18171 ;          8289  !       WHEN STRIPPING AWAY BAD NIBBLE
18172 ;          8290  !       LOCATIONS OF A BLOCK.
18173 ;          8291  !
18174 ;          8292  !
```

```
18176 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
18177 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (88)
18178
18179 ;          8293  !--
18180 ;          8294
18181 ;          8295  local
18182 ;          8296      SAV_NIB,                           !STORES THE SHIFTED BIT PATTERN
18183 ;          8297      DODU_FLG,                          !DROP UNIT FLAG
18184 ;          8298      NIB_BIT,                           !NIBBLE PATTERN
18185 ;          8299      TST_PAT,                           !TEST PATTERN
18186 ;          8300      ALL_ONES_1,                        !HOW MANY '17' NIBBLE PATTERN POSITION TO READ
18187 ;          8301      ALL_ONES_2,                        !HOW MANY '17' NIBBLE PATTERN POSITION TO READ
18188 ;          8302      STK_PTR,                           !STACK POINTER
18189 ;          8303      COUNT;                             !COUNTER
18190 ;          8304
18191 ;          8305  BGNSUB;
18192 ;          8306  CLR_MBUS;
18193 ;          8307  DODU_FLG = ZERO;
18194 ;          8308  TST_PAT = ONE;                         !ONE IN A FIELD OF ZEROES
18195 ;          8309
18196 ;          8310  incr CNT from 0 to 15 do               !WRITE 16 WORDS WITH SHIFTING 0 IN FIELD OF 1'S.
18197 ;          8311      begin
18198 ;          8312      IO_BUF [.CNT] = not .TST_PAT;
18199 ;          8313      TST_PAT = .TST_PAT^ONE;
18200 ;          8314      end;
18201 ;          8315
18202 ;          8316  GD_BLK_XFER ();                        !SET UP A GOOD BLOCK XFERR
18203 ;          8317  MLCS1 = write;                         !WRITE SHIFTING PATTERN THROUGH SYNC BUS
18204 ;          8318  TIME_OUT_LOOP;
18205 ;          8319  STRIPPER (21, 8);                      !CALL ROUTINE TO STRIP OUT BAD NIBBLE DATA FROM ARRAY WORDS
18206 ;          8320  STK_PTR = -1;                          !RESET THE STACK POINTER
18207 ;          8321  NIB_BIT = ONE;                         !SHIFTING NIBBLE PAT OF 1 IN FIELD OF 0'S
18208 ;          8322  ALL_ONES_1 = ZERO;                    !READ NO '17' NIBBLE PATTERN ON FIRST PASS
18209 ;          8323  ALL_ONES_2 = 3;                       !READ THREE '17' NIBBLE PATTERN ON FIRST PASS
18210 ;          8324
18211 ;          8325  incr BY_FOUR_WRDS from 0 to 3 do      !READ 4 GROUPS OF 4 WORDS
18212 ;          8326      begin
18213 ;          8327
18214 ;          8328      incr BY_ONE_WRD from 0 to 3 do     !READ 4 GROUPS OF 1 WORD
18215 ;          8329          begin
18216 ;          8330          COUNT = ZERO;                  !CLEAR COUNT
18217 ;          8331
18218 ;          8332          until .COUNT eql .ALL_ONES_1 do    !READ X NUMBER OF '17' NIBBLE PAT
18219 ;          8333              begin
18220 ;          8334              COUNT = .COUNT + 1;        !INCREMENT COUNT
18221 ;          8335              STK_PTR = .STK_PTR + 1;    !INCREMENT STACK POINTER
18222 ;          8336
18223 ;          8337              if (.stack [.STK_PTR]) neq %o'000017'      !COMPARE STACK WITH '17'
18224 ;          8338              then
18225 ;          8339                  begin                  !ERROR AND SET DODU_FLG IF NEQ
18226 ;          8340                  ERRDF (90, SYNC, DUMPER);
18227 ;          8341                  PRINTB (SIX_FMT, WRD_23, WRD_39, PHR_4, WRD_12, FNC_5, WRD_19);
18228 ;          8342                  PRINTB (FMT_5, ONES, .stack [.STK_PTR], .STR_PTR);
18229 ;          8343                  DODU_FLG = ONE;
18230 ;          8344                  end;
```

```
18232 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
18233 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (88)
18234
18235 ;          8345
18236 ;          8346          end;
18237 ;          8347
18238 ;          8348     STK_PTR = .STK_PTR + 1;                    !INCREMENT THE STACK POINTER
18239 ;          8349     SAV_NIB = ( not .NIB_BIT) and (%o'000017');     !GENERATE THE SHIFTED BIT
18240 ;          8350
18241 ;          8351     if (.stack [.STK_PTR]) neq (.SAV_NIB)      !COMPARE STACK TO SHIFTED BIT
18242 ;          8352     then
18243 ;          8353          begin                                 !ERROR AND SET DODU_FLG IF NEQ
18244 ;          8354          ERRDF (91, SYNC, DUMPER);
18245 ;          8355          PRINTB (SIX_FMT, WRD_23, WRD_39, PHR_4, WRD_12, FNC_5, WRD_19);
18246 ;          8356          PRINTB (FMT_15, .STK_PTR);
18247 ;          8357          PRINTB (FMT_5, .SAV_NIB, .stack [.STK_PTR]);
18248 ;          8358          DODU_FLG = ONE;
18249 ;          8359          end;
18250 ;          8360
18251 ;          8361     COUNT = ZEROES;                            !CLEAR COUNT
18252 ;          8362
18253 ;          8363     until .COUNT eql .ALL_ONES_2 do            !READ X NUMBER OF '17' NIBBLE PAT
18254 ;          8364          begin
18255 ;          8365          COUNT = .COUNT + 1;                   !INCREMENT COUNT
18256 ;          8366          STK_PTR = .STK_PTR + 1;               !INCREMENT STACK POINTER
18257 ;          8367
18258 ;          8368          if (.stack [.STK_PTR]) neq %o'000017'      !COMPARE STACK POINTER WITH  '17'
18259 ;          8369          then
18260 ;          8370               begin                            !ERROR AND SET DODU_FLG IF SET
18261 ;          8371               ERRDF (92, SYNC, DUMPER);
18262 ;          8372               PRINTB (SIX_FMT, WRD_23, WRD_39, PHR_4, WRD_12, FNC_5, WRD_19);
18263 ;          8373               PRINTB (FMT_5, ONES, .stack [.STK_PTR], .STK_PTR);
18264 ;          8374               DODU_FLG = ONE;
18265 ;          8375               end;
18266 ;          8376
18267 ;          8377               end;
18268 ;          8378
18269 ;          8379          NIB_BIT = .NIB_BIT^ONE;                !SHIFT THE SHIFTED NIBBLE BIT
18270 ;          8380          end;
18271 ;          8381
18272 ;          8382     NIB_BIT = ONE;                             !RESET THE SHIFTED NIBBLE BIT
18273 ;          8383     ALL_ONES_1 = .ALL_ONES_1 + 1;              !READ ONE MORE '17' PATTERN
18274 ;          8384     ALL_ONES_2 = .ALL_ONES_2 - 1;              !READ ONE LESS '17' PATTERN
18275 ;          8385     end;
18276 ;          8386
18277 ;          8387 ENDSUB;
18278 ;          8388
18279 ;          8389 if .DODU_FLG IS_SET                            !DROP THIS UNIT IF DODU_FLG SET
18280 ;          8390 then
18281 ;          8391     begin
18282 ;          8392     DODU (.ML_LUN);
18283 ;          8393     DOCLN;
18284 ;          8394     end;
18285 ;          8395
18286 ;          8396 ENDTST;
```

```
18288 :ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
18289 :                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (88)
18290
18294
18295                                          .SBTTL  $T38
18299 064062  004167  120030          $T38:   JSR     R1,$SAVE5                       ;                              8239
18300 064066  162706  000012                  SUB     #12,SP
18301 064072  104402                  1$:     TRAP    2                               ;                              8303
18302 064074  152777  000040  127112          BISB    #40,@ML.REG+40                  ;                              8305
18303 064102  016705  127474                  MOV     ML.DUT,R5
18304 064106  042705  177770                  BIC     #177770,R5
18305 064112  142777  000007  127074          BICB    #7,@ML.REG+40
18306 064120  150577  127070                  BISB    R5,@ML.REG+40
18307 064124  005001                          CLR     R1                              ; DODU.FLG                     8307
18308 064126  012766  000001  000006          MOV     #1,6(SP)                        ; *,TST.PAT                    8308
18309 064134  005004                          CLR     R4                              ; CNT                          8310
18310 064136  010405                  2$:     MOV     R4,R5                           ; CNT,*                        8312
18311 064140  006305                          ASL     R5
18312 064142  016665  000006  011600          MOV     6(SP),IO.BUF(R5)                ; TST.PAT,*
18313 064150  005165  011600                  COM     IO.BUF(R5)
18314 064154  006366  000006                  ASL     6(SP)                           ; TST.PAT                      8313
18315 064160  005204                          INC     R4                              ; CNT                          8310
18316 064162  020427  000017                  CMP     R4,#17                          ; CNT,*
18317 064166  003763                          BLE     2$
18318 064170  004767  131674                  JSR     PC,GD.BLK.XFER                  ;                              8316
18319 064174  012777  000061  126752          MOV     #61,@ML.REG                     ;                              8317
18320 064202  105777  127016          3$:     TSTB    @ML.REG+50
18321 064206  100375                          BPL     3$
18322 064210  012746  000025                  MOV     #25,-(SP)                       ;                              8319
18323 064214  012746  000010                  MOV     #10,-(SP)
18324 064220  004767  127770                  JSR     PC,STRIPPER
18325 064224  012702  177777                  MOV     #-1,R2                          ; *,STK.PTR                    8320
18326 064230  012766  000001  000004          MOV     #1,4(SP)                        ; *,NIB.BIT                    8321
18327 064236  005066  000010                  CLR     10(SP)                          ; ALL.ONES.1                   8322
18328 064242  012766  000003  000006          MOV     #3,6(SP)                        ; *,ALL.ONES.2                 8323
18329 064250  005004                          CLR     R4                              ; BY.FOUR.WRDS                 8325
18330 064252  005005                  4$:     CLR     R5                              ; BY.ONE.WRD                   8328
18331 064254  005003                  5$:     CLR     R3                              ; COUNT                        8330
18332 064256  020366  000010          6$:     CMP     R3,10(SP)                       ; COUNT,ALL.ONES.1             8332
18333 064262  001455                          BEQ     7$
18334 064264  005203                          INC     R3                              ; COUNT                        8334
18335 064266  005202                          INC     R2                              ; STK.PTR                      8335
18336 064270  126227  012612  000017          CMPB    STACK(R2),#17                   ; *(STK.PTR),*                 8337
18337 064276  001767                          BEQ     6$
18338 064300  104455                          TRAP    55                              ;                              8340
18339 064302  000132                          .WORD   132
18340 064304  010526                          .WORD   SYNC
18341 064306  024060                          .WORD   DUMPER
18342 064310  012746  006506                  MOV     #WRD.19,-(SP)                   ;                              8341
```

```
                                        ;ML4                                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                                       14-May-1981 13:05:08    PA:<
18347 064314  012746  007774           MOV     #FNC.5,-(SP)
18348 064320  012746  006426           MOV     #WRD.12,-(SP)
18349 064324  012746  007454           MOV     #PHR.4,-(SP)
18350 064330  012746  006716           MOV     #WRD.39,-(SP)
18351 064334  012746  006544           MOV     #WRD.23,-(SP)
18352 064340  012746  006100           MOV     #SIX.FMT,-(SP)
18353 064344  012746  000007           MOV     #7,-(SP)
18354 064350  010600                   MOV     SP,R0                   ; SP,*
18355 064352  104414                   TRAP    14
18356 064354  010216                   MOV     R2,(SP)                 ; STK.PTR,*                             8342
18357 064356  005046                   CLR     -(SP)
18358 064360  116216  012612           MOVB    STACK(R2),(SP)          ; *(STK.PTR),*
18359 064364  012746  177777           MOV     #-1,-(SP)
18360 064370  012746  004430           MOV     #FMT.5,-(SP)
18361 064374  012746  000004           MOV     #4,-(SP)
18362 064400  010600                   MOV     SP,R0                   ; SP,*
18363 064402  104414                   TRAP    14
18364 064404  012701  000001           MOV     #1,R1                   ; *,DODU.FLG                            8343
18365 064410  062706  000030           ADD     #30,SP                  ;                                       8339
18366 064414  000720                   BR      6$                      ;                                       8332
18367 064416  005202           7$:     INC     R2                      ; STK.PTR                               8348
18368 064420  012766  000017  000014   MOV     #17,14(SP)              ; *,SAV.NIB                             8349
18369 064426  046666  000004  000014   BIC     4(SP),14(SP)            ; NIB.BIT,SAV.NIB
18370 064434  005000                   CLR     R0                      ;                                       8351
18371 064436  156200  012612           BISB    STACK(R2),R0            ; *(STK.PTR),*
18372 064442  020066  000014           CMP     R0,14(SP)               ; *,SAV.NIB
18373 064446  001454                   BEQ     8$
18374 064450  104455                   TRAP    55                      ;                                       8354
18375 064452  000133                   .WORD   133
18376 064454  010526                   .WORD   SYNC
18377 064456  024060                   .WORD   DUMPER
18378 064460  012746  006506           MOV     #WRD.19,-(SP)           ;                                       8355
18379 064464  012746  007774           MOV     #FNC.5,-(SP)
18380 064470  012746  006426           MOV     #WRD.12,-(SP)
18381 064474  012746  007454           MOV     #PHR.4,-(SP)
18382 064500  012746  006716           MOV     #WRD.39,-(SP)
18383 064504  012746  006544           MOV     #WRD.23,-(SP)
18384 064510  012746  006100           MOV     #SIX.FMT,-(SP)
18385 064514  012746  000007           MOV     #7,-(SP)
18386 064520  010600                   MOV     SP,R0                   ; SP,*
18387 064522  104414                   TRAP    14
18388 064524  010216                   MOV     R2,(SP)                 ; STK.PTR,*                             8356
18389 064526  012746  005150           MOV     #FMT.15,-(SP)
18390 064532  012746  000002           MOV     #2,-(SP)
18391 064536  010600                   MOV     SP,R0                   ; SP,*
18392 064540  104414                   TRAP    14
18393 064542  005016                   CLR     (SP)                    ;                                       8357
18394 064544  116216  012612           MOVB    STACK(R2),(SP)          ; *(STK.PTR),*
18395 064550  016646  000040           MOV     40(SP),-(SP)            ; SAV.NIB,*
18396 064554  012746  004430           MOV     #FMT.5,-(SP)
18397 064560  012746  000003           MOV     #3,-(SP)
18398 064564  010600                   MOV     SP,R0                   ; SP,*
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        ;                                                       14-May-1981 13:05:08    PA:<
                                        TRAP    14
064566  104414
064570  012701  000001                  MOV     #1,R1           ; *,DODU.FLG                                    8358
064574  062706  000032                  ADD     #32,SP          ;                                               8353
064600  005003                  8$:     CLR     R3              ; COUNT                                         8361
064602  020366  000006          9$:     CMP     R3,6(SP)        ; COUNT,ALL.ONES.2                              8363
064606  001455                          BEQ     10$
064610  005203                          INC     R3              ; COUNT                                         8365
064612  005202                          INC     R2              ; STK.PTR                                       8366
064614  126227  012612  000017          CMPB    STACK(R2),#17   ; *(STK.PTR),*                                  8368
064622  001767                          BEQ     9$
064624  104455                          TRAP    55              ;                                               8371
064626  000134                          .WORD   134
064630  010526                          .WORD   SYNC
064632  024060                          .WORD   DUMPER
064634  012746  006506                  MOV     #WRD.19,-(SP)   ;                                               8372
064640  012746  007774                  MOV     #FNC.5,-(SP)
064644  012746  006426                  MOV     #WRD.12,-(SP)
064650  012746  007454                  MOV     #PHR.4,-(SP)
064654  012746  006716                  MOV     #WRD.39,-(SP)
064660  012746  006544                  MOV     #WRD.23,-(SP)
064664  012746  006100                  MOV     #SIX.FMT,-(SP)
064670  012746  000007                  MOV     #7,-(SP)
064674  010600                          MOV     SP,R0           ; SP,*
064676  104414                          TRAP    14
064700  010216                          MOV     R2,(SP)         ; STK.PTR,*                                     8373
064702  005046                          CLR     -(SP)
064704  116216  012612                  MOVB    STACK(R2),(SP)  ; *(STK.PTR),*
064710  012746  177777                  MOV     #-1,-(SP)
064714  012746  004430                  MOV     #FMT.5,-(SP)
064720  012746  000004                  MOV     #4,-(SP)
064724  010600                          MOV     SP,R0           ; SP,*
064726  104414                          TRAP    14
064730  012701  000001                  MOV     #1,R1           ; *,DODU.FLG                                    8374
064734  062706  000030                  ADD     #30,SP          ;                                               8370
064740  000720                          BR      9$              ;                                               8363
064742  006366  000004          10$:    ASL     4(SP)           ; NIB.BIT                                       8379
064746  005205                          INC     R5              ; BY.ONE.WRD                                    8328
064750  020527  000003                  CMP     R5,#3           ; BY.ONE.WRD,*
064754  003002                          BGT     11$
064756  000167  177272                  JMP     5$
064762  012766  000001  000004  11$:    MOV     #1,4(SP)        ; *,NIB.BIT                                     8392
064770  005266  000010                  INC     10(SP)          ; ALL.ONES.1                                    8383
064774  005366  000006                  DEC     6(SP)           ; ALL.ONES.2                                    8384
065000  005204                          INC     R4              ; BY.FOUR.WRDS                                  8325
065002  020427  000003                  CMP     R4,#3           ; BY.FOUR.WRDS,*
065006  003002                          BGT     12$
065010  000167  177236                  JMP     4$
065014  022626                  12$:    CMP     (SP)+,(SP)+     ;                                               8303
065016  104467                          TRAP    67              ;                                               8385
065020  006000                          ROR     R0
065022  103002                          BHIS    13$
065024  000167  177042                  JMP     1$
```

```
18456                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
18457                                   ;                                                       14-May-1981 13:05:08    PA:<
18458
18459 065030  005301                    13$:    DEC     R1                      ; DODU.FLG                      8389
18460 065032  001004                            BNE     14$
18461 065034  016700  126540                    MOV     ML.LUN,R0               ;                               8392
18462 065040  104451                            TRAP    51
18463 065042  104444                            TRAP    44
18464 065044  062706  000012              14$:    ADD     #12,SP                  ;                               8239
18465 065050  000207                            RTS     PC
18466
18467                                   ; Routine Size:  252 words
18468                                   ; Maximum stack depth per invocation:  26 words
18473
18474
18478
18479                                           .SBTTL  T38
18483 065052                            T38::
18484 065052  004767  177004            1$:     JSR     PC,$T38                 ;                               8394
18485 065056  104466                            TRAP    66
18486 065060  006000                            ROR     R0
18487 065062  103773                            BLO     1$
18488 065064  000207                            RTS     PC
18489
18490                                   ; Routine Size:  6 words
18491                                   ; Maximum stack depth per invocation:  0 words
18496
18497
18498 ;          8397  !<BLF/PAGE>
```

```
18500 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
18501 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (89)
18502
18503 ;          8398  !
18504 ;          8399  BGNTST;
18505 ;          8400
18506 ;          8401  !++
18507 ;          8402  ! TEST NUMBER:  TST 39
18508 ;          8403  !
18509 ;          8404  ! TEST NAME:    SYNC DATA BUS BIT UNIQUENESS TEST/READ PATH
18510 ;          8405  !
18511 ;          8406  ! TEST DESCRIPTION:
18512 ;          8407  !       TEST SYNCHRONOUS DATA BUS READ
18513 ;          8408  !       PATH FOR DATA BIT UNIQUENESS BY:
18514 ;          8409  !
18515 ;          8410  !       1.  LOADING THE FIRST 16 WORDS IN THE IO_BUF WITH A SHIFTING
18516 ;          8411  !           ZERO IN A FIELD OF ONES PATTERN.
18517 ;          8412  !
18518 ;          8413  !       2.  VIA MBUS WRITE FUNCTION WRITE SHIFTING PATTERN INTO THE GOOD BLOCK.
18519 ;          8414  !
18520 ;          8415  !       3.  CLEAR THE IO_BUF.
18521 ;          8416  !
18522 ;          8417  !       4.  VIA MBUS READ FUNCTION READ THE SHIFTING PATTERN THROUGH THE
18523 ;          8418  !           READ PATH.
18524 ;          8419  !
18525 ;          8420  !       5.  INTERIGATE THE IO_BUF FOR THE SHIFTING PATTERN.
18526 ;          8421  !
18527 ;          8422  ! IMPLICIT INPUTS:
18528 ;          8423  !       IO_BUF
18529 ;          8424  !       A VECTOR OF 256 WORDS WHERE DATA FOR MBUS READS AND WRITE FUNCTION ARD FOUND.
18530 ;          8425  !--
18531 ;          8426
18532 ;          8427  local
18533 ;          8428      DODU_FLG,                               !DROP UNIT FLAG
18534 ;          8429      TST_PAT;                                !TEST PATTERN
18535 ;          8430
18536 ;          8431  CLR_MBUS;
18537 ;          8432  DODU_FLG = ZERO;
18538 ;          8433  TST_PAT = ONE;                              !ONE IN A FIELD OF ZEROES
18539 ;          8434
18540 ;          8435  incr CNT from 0 to 15 do                    !WRITE 16 WORDS WITH SHIFTED 0 IN A FIELD OF 1'S
18541 ;          8436      begin
18542 ;          8437      IO_BUF [.CNT] = not .TST_PAT;
18543 ;          8438      TST_PAT = .TST_PAT^ONE;
18544 ;          8439      end;
18545 ;          8440
18546 ;          8441  GD_BLK_XFER ();                             !SET UP A GOOD BLOCK XFERR
18547 ;          8442  ML[S1 = write;                              !WRITE SHIFTING PATTERN
18548 ;          8443  TIME_OUT_LOOP;
18549 ;          8444  BGNSUB;
18550 ;          8445
18551 ;          8446  incr CNT from 0 to 15 do                    !CLEAR OUT THE IO_BUF
18552 ;          8447      IO_BUF [.CNT] = ZEROES;
18553 ;          8448
18554 ;          8449  CLR_MBUS;
```

```
18556 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
18557 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (89)
18558
18559 ;          8450  GD_BLK_XFER ();                        !SET UP A GOOD BLOCK XFERR
18560 ;          8451  ML[CS1 = read;                         !READ SHIFTING PATTERN THROUGH SYNC BUS
18561 ;          8452  TIME_OUT_LOOP;
18562 ;          8453  TST_PAT = ONE;                         !SHIFTING PATTERN
18563 ;          8454
18564 ;          8455  incr CNT from 0 to 15 do               !READ IO_BUF FOR SHIFTING 0 IN FIELD OF 1'S
18565 ;          8456      begin
18566 ;          8457
18567 ;          8458      if .IO_BUF [.CNT] neq ( not .TST_PAT)   !COMPARE IO_BUF TO SHIFTED PAT
18568 ;          8459      then
18569 ;          8460          begin                          !ERROR AND SET DODU_FLG IF NEQ
18570 ;          8461          ERRDF (93, SYNC, DUMPER);
18571 ;          8462          PRINTB (FIV_FMT, WRD_23, FNC_6, WRD_19, WRD_39, PHR_4);
18572 ;          8463          PRINTB (FMT_2, ( not .TST_PAT), .IO_BUF [.CNT], ( not .TST_PAT xor .IO_BUF [.CNT]));
18573 ;          8464          DODU_FLG = ONE;
18574 ;          8465          end;
18575 ;          8466
18576 ;          8467      TST_PAT = .TST_PAT^ONE;            !SHIFT THE PATTERN AND REPEAT
18577 ;          8468      end;
18578 ;          8469
18579 ;          8470  ENDSUB;
18580 ;          8471
18581 ;          8472  if .DODU_FLG IS_SET                    !DROP THIS UNIT IF DODU_FLG IS_SET
18582 ;          8473  then
18583 ;          8474      begin
18584 ;          8475      DODU (.ML_LUN);
18585 ;          8476      DOCLN;
18586 ;          8477      end;
18587 ;          8478
18588 ;          8479  ENDTST;
18592
18593                                        .SBTTL  $T39
18597 065066  004167  117024          $T39: JSR     R1,$SAVE5          ;                          8396
18598 065072  152777  000040  126114        BISB    #40,@ML.REG+40     ;                          8429
18599 065100  016705  126476                MOV     ML.DUT,R5
18600 065104  042705  177770                BIC     #177770,R5
18601 065110  142777  000007  126076        BICB    #7,@ML.REG+40
18602 065116  150577  126072                BISB    R5,@ML.REG+40
18603 065122  005005                        CLR     R5                 ; DODU.FLG             8432
18604 065124  012704  000001                MOV     #1,R4              ; *,TST.PAT            8433
18605 065130  005000                        CLR     R0                 ; CNT                  8435
18606 065132  010001                  1$:   MOV     R0,R1              ; CNT,*                8437
18607 065134  006301                        ASL     R1
18608 065136  010461  011600                MOV     R4,IO.BUF(R1)      ; TST.PAT,*
18609 065142  005161  011600                COM     IO.BUF(R1)
18610 065146  006304                        ASL     R4                 ; TST.PAT              8438
```

```
                                                                                    14-May-1981 13:48:54    TOPS
                                                                                    14-May-1981 13:05:08    PA:<
18612                           ;ML4
18613                           ;
18614
18615 065150  005200                    INC     R0                      ; CNT                           8435
18616 065152  020027  000017              CMP     R0,#17                  ; CNT,*
18617 065156  003765                      BLE     1$
18618 065160  004767  130704              JSR     PC,GD.BLK.XFER          ;                               8441
18619 065164  012777  000061  125762      MOV     #61,@ML.REG             ;                               8442
18620 065172  105777  126026      2$:     TSTB    @ML.REG+50
18621 065176  100375                      BPL     2$
18622 065200  104402              3$:     TRAP    2                       ;                               8443
18623 065202  005000                      CLR     R0                      ; CNT                           8446
18624 065204  010001              4$:     MOV     R0,R1                   ; CNT,*                         8447
18625 065206  006301                      ASL     R1
18626 065210  005061  011600              CLR     IO.BUF(R1)
18627 065214  005200                      INC     R0                      ; CNT                           8446
18628 065216  020027  000017              CMP     R0,#17                  ; CNT,*
18629 065222  003770                      BLE     4$
18630 065224  152777  000040  125762      BISB    #40,@ML.REG+40          ;                               8447
18631 065232  016703  126344              MOV     ML.DUT,R3
18632 065236  042703  177770              BIC     #177770,R3
18633 065242  142777  000007  125744      BICB    #7,@ML.REG+40
18634 065250  150377  125740              BISB    R3,@ML.REG+40
18635 065254  004767  130610              JSR     PC,GD.BLK.XFER          ;                               8450
18636 065260  012777  000071  125666      MOV     #71,@ML.REG             ;                               8451
18637 065266  105777  125732      5$:     TSTB    @ML.REG+50
18638 065272  100375                      BPL     5$
18639 065274  012704  000001              MOV     #1,R4                   ; *,TST.PAT                     8453
18640 065300  005002                      CLR     R2                      ; CNT                           8455
18641 065302  010201              6$:     MOV     R2,R1                   ; CNT,*                         8458
18642 065304  006301                      ASL     R1
18643 065306  012703  011600              MOV     #IO.BUF,R3
18644 065312  060103                      ADD     R1,R3
18645 065314  010401                      MOV     R4,R1                   ; TST.PAT,*
18646 065316  005101                      COM     R1
18647 065320  021301                      CMP     (R3),R1
18648 065322  001447                      BEQ     7$
18649 065324  104455                      TRAP    55                      ;                               8461
18650 065326  000135                      .WORD   135
18651 065330  010526                      .WORD   SYNC
18652 065332  024060                      .WORD   DUMPER
18653 065334  012746  007454              MOV     #PHR.4,-(SP)            ;                               8462
18654 065340  012746  006716              MOV     #WRD.39,-(SP)
18655 065344  012746  006506              MOV     #WRD.19,-(SP)
18656 065350  012746  010004              MOV     #FNC.6,-(SP)
18657 065354  012746  006544              MOV     #WRD.23,-(SP)
18658 065360  012746  006062              MOV     #FIV.FMT,-(SP)
18659 065364  012746  000006              MOV     #6,-(SP)
18660 065370  010600                      MOV     SP,R0                   ; SP,*
18661 065372  104414                      TRAP    14
18662 065374  011316                      MOV     (R3),(SP)               ;                               8463
18663 065376  010146                      MOV     R1,-(SP)
18664 065400  046616  000002              BIC     2(SP),(SP)
18665 065404  040166  000002              BIC     R1,2(SP)
18666 065410  052616                      BIS     (SP)+,(SP)
```

```
18668                           ;ML4                                                    14-May-1981 13:48:54    TOPS
18669                           ;                                                       14-May-1981 13:05:08    PA:<
18670
18671 065412  011346                     MOV     (R3),-(SP)
18672 065414  010146                     MOV     R1,-(SP)
18673 065416  012746  004266             MOV     #FMT.2,-(SP)
18674 065422  012746  000004             MOV     #4,-(SP)
18675 065426  010600                     MOV     SP,R0                          ; SP,*
18676 065430  104414                     TRAP    14
18677 065432  012705  000001             MOV     #1,R5                          ; *,DODU.FLG                    8464
18678 065436  062706  000026             ADD     #26,SP                         ;                               8460
18679 065442  006304            7$:      ASL     R4                             ; TST.PAT                       8467
18680 065444  005202                     INC     R2                             ; CNT                           8455
18681 065446  020227  000017             CMP     R2,#17                         ; CNT,*
18682 065452  003713                     BLE     6$
18683 065454  104467                     TRAP    67                             ;                               8468
18684 065456  006000                     ROR     R0
18685 065460  103647                     BLO     3$
18686 065462  005305                     DEC     R5                             ; DODU.FLG                      8472
18687 065464  001004                     BNE     8$
18688 065466  016700  126106             MOV     ML.LUN,R0                      ;                               8475
18689 065472  104451                     TRAP    51
18690 065474  104444                     TRAP    44
18691 065476  000207            8$:      RTS     PC                             ;                               8396
18692
18693                           ; Routine Size:  133 words
18694                           ; Maximum stack depth per invocation:  17 words
18699
18700
18704
18705                                   .SBTTL  T39
18709 065500                    T39::
18710 065500  004767  177362    1$:      JSR     PC,$T39                        ;                               8477
18711 065504  104466                     TRAP    66
18712 065506  006000                     ROR     R0
18713 065510  103773                     BLO     1$
18714 065512  000207                     RTS     PC
18715
18716                           ; Routine Size:  6 words
18717                           ; Maximum stack depth per invocation:  0 words
18722 ;         8480  !<BLF/PAGE>
```

```
18724 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
18725 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (90)
18726
18727 ;          8481  !
18728 ;          8482  !
18729 ;          8483  BGNTST;
18730 ;          8484
18731 ;          8485  !++
18732 ;          8486  ! TEST NUMBER:  TST 40
18733 ;          8487  !
18734 ;          8488  ! TEST NAME:    ARRAY ADDRESS MUX TEST
18735 ;          8489  !
18736 ;          8490  ! TEST DESCRIPTION:
18737 ;          8491  !
18738 ;          8492  !       TEST FOR UNIQUE MOS RAM ROW
18739 ;          8493  !       AND COLUMN ADDRESSING BY:
18740 ;          8494  !
18741 ;          8495  !       1.  FIRST FINDING A ERROR FREE
18742 ;          8496  !           16K OR 64K CHUNK OF MEMORY.
18743 ;          8497  !           THIS REPRESENTS ONE ROW OF
18744 ;          8498  !           EITHER 16K OR 64K MOS RAMS.
18745 ;          8499  !
18746 ;          8500  !       2.  WRITE A BACKGROUND OF ALL
18747 ;          8501  !           ONES INTO THE GOOD CHUNK
18748 ;          8502  !
18749 ;          8503  !       3.  WRITE ZEROES INTO THE FIRST
18750 ;          8504  !           BLOCK OF THE GOOD CHUNK.
18751 ;          8505  !
18752 ;          8506  !       4.  READ REMAINING BLOCKS IN
18753 ;          8507  !           GOOD CHUNK FOR ONES.
18754 ;          8508  !
18755 ;          8509  !
18756 ;          8510  ! IMPLICIT INPUTS:
18757 ;          8511  !
18758 ;          8512  !       IO_BUF
18759 ;          8513  !
18760 ;          8514  !       A VECTOR OF 256 WORDS
18761 ;          8515  !       WHERE DATA FOR MBUS
18762 ;          8516  !       READ AND WRITE TRANSFERS
18763 ;          8517  !       CAN BE FOUND.
18764 ;          8518  !--
18765 ;          8519
18766 ;          8520  local
18767 ;          8521      DSA_ADRS,                       !DSA ADRS COUNTER
18768 ;          8522      FND_GD_CHK;                     !FOUND GOOD 16K/64K CHUNK FLAG
18769 ;          8523
18770 ;          8524  IO_BUF = ONES;                      !LOAD FIRST IO_BUF WORD WITH ONES
18771 ;          8525  DSA_ADRS = -.RAS_INC;               !REST DSA COUNT
18772 ;          8526  FND_GD_CHK = ZERO;                  !CLEAR FLAG
18773 ;          8527
18774 ;          8528  do                                  !DO UNTIL FOUND GOOD CHUNK OR LBT
18775 ;          8529      begin
18776 ;          8530      CLR_MBUS;
18777 ;          8531      BAI = ONE;                      !SET ON FIRST IO BUF WORD
18778 ;          8532      ECC_DIS = ONE;                  !DISABLE ECC
```

```
18780 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
18781 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (90)
18782
18783 ;          8533      DSA_ADRS = .DSA_ADRS + .RAS_INC;          !INCREMENT DSA ADRS COUNTER
18784 ;          8534      MLWC = .W_C_SIZE;                         !16K OR 64K WORDS
18785 ;          8535      MLBA = IO_BUF;                            !LOAD UBUS ADRS
18786 ;          8536      MLDA = .DSA_ADRS;                         !LOAD DSA ADRS
18787 ;          8537      ML_FUNC = write;                          !DO A WRITE FUNCTION
18788 ;          8538      TIME_OUT_LOOP;
18789 ;          8539
18790 ;          8540      if .SC IS_NOT_SET                         !DID XFERR CAUSE AN SC
18791 ;          8541      then
18792 ;          8542          begin                                 !XFERR WAS OK
18793 ;          8543          MLWC = .W_C_SIZE;                     !LOAD WORD COUNT
18794 ;          8544          MLBA = IO_BUF;                        !LOAD UBUS ADRS
18795 ;          8545          MLDA = .DSA_ADRS;                     !LOAD DSA ADRS
18796 ;          8546          ML_FUNC = WRT_CHK;                    !DO A WRITE CHECK FUNCTION
18797 ;          8547          TIME_OUT_LOOP;
18798 ;          8548
18799 ;          8549          if .SC IS_NOT_SET                     !IS THIS CHUNK GOOD
18800 ;          8550          then
18801 ;          8551              FND_GD_CHK = ONE;                 !YES SET FLG
18802 ;          8552
18803 ;          8553          end
18804 ;          8554
18805 ;          8555      end
18806 ;          8556  until (.FND_GD_CHK IS_SET ) or (.DSA_ADRS eql .LST_ARR + .ARR_INC);
18807 ;          8557
18808 ;          8558                                                !REPEAT UNTIL FOUND GOOD CHUNCK OR AT LBT
18809 ;          8559
18810 ;          8560  if .DSA_ADRS eql .LST_ARR + .ARR_INC
18811 ;          8561  then
18812 ;          8562      begin
18813 ;          8563      ERRDF (111, INTER, DUMPER);
18814 ;          8564      PRINTB (FIV_FMT, FNC_13, FNC_17, WRD_50, WRD_60, WRD_56);
18815 ;          8565      PRINTB (THR_FMT, WRD_14, PHR_10, FNC_15);
18816 ;          8566      DODU (.ML_LUN);
18817 ;          8567      DOCLN;
18818 ;          8568      end
18819 ;          8569  else                                          !A GOOD CHUNCK WAS FOUND CONTINUE TEST
18820 ;          8570      begin
18821 ;          8571      CLR_MBUS;
18822 ;          8572      BAI = ONE;
18823 ;          8573      ECC_DIS = ONE;                            !DISABLE ECC
18824 ;          8574      IO_BUF = ZEROES;                          !FIRST BLOCK IN CHUNCK GETS ZEROES
18825 ;          8575      MLDA = .DSA_ADRS;                         !LOAD DSA
18826 ;          8576      MLWC = not 255;                           !LOAD WORD COUNT
18827 ;          8577      MLBA = IO_BUF;                            !LOAD UBUS ADRS
18828 ;          8578      ML_FUNC = write;                          !DO A WRITE FUNCTION
18829 ;          8579      TIME_OUT_LOOP;
18830 ;          8580      CLR_MBUS;
18831 ;          8581      BAI = ONE;
18832 ;          8582      IO_BUF = ONES;                            !READ REMAINING BLOCKS FOR ONES
18833 ;          8583      ECC_DIS = ONE;
18834 ;          8584      MLDA = .DSA_ADRS + 1;                     !LOAD DSA WITH NEXT DSA INCREMENT
```

```
18836 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
18837 .                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (90)
18838
18839 ;          8585      MLBA = IO_BUF;                     !LOAD  UBUS ADRS
18840 ;          8586      MLWC = .W_C_SIZE + 256;            !LOAD WORD COUNT WITH LESS ONE BLOCK
18841 ;          8587      ML_FUNC = WRT_CHK;                 !DO A WRITE CHECK FUNCTION
18842 ;          8588      TIME_OUT_LOOP;
18843 ;          8589
18844 ;          8590      if .WCE IS_SET                     !WERE ANY BITS DISTURBED
18845 ;          8591      then
18846 ;          8592          begin                          !ERROR IF WRITE CHECK FOUND BAD DATA
18847 ;          8593          ERRDF (112, ASYNC, DUMPER);
18848 ;          8594          PRINTB (FOR_FMT, FNC_17, WRD_50, WRD_60, WRD_14);
18849 ;          8595          PRINTB (FMT_9, (.MLDA - 1));
18850 ;          8596          DODU (.ML_LUN);
18851 ;          8597          DOCLN;
18852 ;          8598          end;
18853 ;          8599
18854 ;          8600      end;
18855 ;          8601
18856 ;          8602  ENDTST;
18860
18861                                    .SBTTL  $T40
18865 065514  004167  116324          $T40:   JSR     R1,$SAVE2                  ;                              8479
18866 065520  012767  177777  124052          MOV     #-1,IO.BUF                 ;                              8524
18867 065526  016702  125372                  MOV     RAS.INC,R2                 ; *,DSA.ADRS                   8525
18868 065532  005402                          NEG     R2                         ; DSA.ADRS
18869 065534  005000                          CLR     R0                         ; FND.GD.CHK                   8526
18870 065536  152777  000040  125450  1$:     BISB    #40,@ML.REG+40             ;                              8529
18871 065544  016701  126032                  MOV     ML.DUT,R1
18872 065550  042701  177770                  BIC     #177770,R1
18873 065554  142777  000007  125432          BICB    #7,@ML.REG+40
18874 065562  150177  125426                  BISB    R1,@ML.REG+40
18875 065566  152777  000010  125420          BISB    #10,@ML.REG+40             ;                              8531
18876 065574  152777  000002  125472          BISB    #2,@ML.REG+120             ;                              8532
18877 065602  066702  125316                  ADD     RAS.INC,R2                 ; *,DSA.ADRS                   8533
18878 065606  016777  125310  125350          MOV     W.C.SIZE,@ML.REG+10        ;                              8534
18879 065614  012777  011600  125352          MOV     #IO.BUF,@ML.REG+20         ;                              8535
18880 065622  010277  125356                  MOV     R2,@ML.REG+30              ; DSA.ADRS,*                   8536
18881 065626  142777  000077  125320          BICB    #77,@ML.REG                ;                              8537
18882 065634  152777  000061  125312          BISB    #61,@ML.REG
18883 065642  105777  125356          2$:     TSTB    @ML.REG+50
18884 065646  100375                          BPL     2$
18885 065650  032777  100000  125276          BIT     #100000,@ML.REG            ;                              8540
18886 065656  001027                          BNE     4$
18887 065660  016777  125236  125276          MOV     W.C.SIZE,@ML.REG+10        ;                              8543
18888 065666  012777  011600  125300          MOV     #IO.BUF,@ML.REG+20         ;                              8544
18889 065674  010277  125304                  MOV     R2,@ML.REG+30              ; DSA.ADRS,*                   8545
18890 065700  142777  000077  125246          BICB    #77,@ML.REG                ;                              8546
```

```
18892                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
18893                                   ;                                                       14-May-1981 13:05:08    PA:<
18894
18895 065706  152777  000051  125240          BISB    #51,@ML.REG
18896 065714  105777  125304          3$:     TSTB    @ML.REG+50
18897 065720  100375                  BPL     3$
18898 065722  032777  100000  125224          BIT     #100000,@ML.REG                 ;                                       8549
18899 065730  001002                  BNE     4$
18900 065732  012700  000001          MOV     #1,R0                           ; *,FND.GD.CHK                          8551
18901 065736  020027  000001          4$:     CMP     R0,#1                           ; FND.GD.CHK,*                          8556
18902 065742  001406                  BEQ     5$
18903 065744  016701  123626          MOV     LST.ARR,R1
18904 065750  066701  123606          ADD     ARR.INC,R1
18905 065754  020201                  CMP     R2,R1                           ; DSA.ADRS,*
18906 065756  001267                  BNE     1$
18907 065760  016701  123612          5$:     MOV     LST.ARR,R1                      ;                                       8560
18908 065764  066701  123572          ADD     ARR.INC,R1
18909 065770  020201                  CMP     R2,R1                           ; DSA.ADRS,*
18910 065772  001047                  BNE     6$
18911 065774  104455                  TRAP    55                              ;                                       8563
18912 065776  000157                  .WORD   157
18913 066000  010672                  .WORD   INTER
18914 066002  024060                  .WORD   DUMPER
18915 066004  012746  007122          MOV     #WRD.56,-(SP)                   ;                                       8564
18916 066010  012746  007156          MOV     #WRD.60,-(SP)
18917 066014  012746  007046          MOV     #WRD.50,-(SP)
18918 066020  012746  010162          MOV     #FNC.17,-(SP)
18919 066024  012746  010112          MOV     #FNC.13,-(SP)
18920 066030  012746  006062          MOV     #FIV.FMT,-(SP)
18921 066034  012746  000006          MOV     #6,-(SP)
18922 066040  010600                  MOV     SP,R0                           ; SP,*
18923 066042  104414                  TRAP    14
18924 066044  012716  010136          MOV     #FNC.15,(SP)                    ;                                       8565
18925 066050  012746  007564          MOV     #PHR.10,-(SP)
18926 066054  012746  006442          MOV     #WRD.14,-(SP)
18927 066060  012746  006034          MOV     #THR.FMT,-(SP)
18928 066064  012746  000004          MOV     #4,-(SP)
18929 066070  010600                  MOV     SP,R0                           ; SP,*
18930 066072  104414                  TRAP    14
18931 066074  016700  125500          MOV     ML.LUN,R0                       ;                                       8566
18932 066100  104451                  TRAP    51
18933 066102  104444                  TRAP    44
18934 066104  062706  000026          ADD     #26,SP                          ;                                       8562
18935 066110  000207                  RTS     PC                              ;                                       8560
18936 066112  152777  000040  125074  6$:     BISB    #40,@ML.REG+40                  ;                                       8570
18937 066120  016701  125456          MOV     ML.DUT,R1
18938 066124  042701  177770          BIC     #177770,R1
18939 066130  142777  000007  125056          BICB    #7,@ML.REG+40
18940 066136  150177  125052          BISB    R1,@ML.REG+40
18941 066142  152777  000010  125044          BISB    #10,@ML.REG+40                  ;                                       8572
18942 066150  152777  000002  125116          BISB    #2,@ML.REG+120                  ;                                       8573
18943 066156  005067  123416          CLR     IO.BUF                          ;                                       8574
18944 066162  010277  125016          MOV     R2,@ML.REG+30                   ; DSA.ADRS,*                            8575
18945 066166  012777  177400  124770          MOV     #-400,@ML.REG+10                ;                                       8576
18946 066174  012777  011600  124772          MOV     #IO.BUF,@ML.REG+20              ;                                       8577
```

```
18948                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
18949                                   ;                                                       14-May-1981 13:05:08    PA:<
18950
18951 066202  142777  000077  124744            BICB    #77,@ML.REG                     ;                               8578
18952 066210  152777  000061  124736            BISB    #61,@ML.REG
18953 066216  105777  125002          7$:       TSTB    @ML.REG+50
18954 066222  100375                            BPL     7$
18955 066224  152777  000040  124762            BISB    #40,@ML.REG+40                  ;                               8579
18956 066232  016701  125344                    MOV     ML.DUT,R1
18957 066236  042701  177770                    BIC     #177770,R1
18958 066242  142777  000007  124744            BICB    #7,@ML.REG+40
18959 066250  150177  124740                    BISB    R1,@ML.REG+40
18960 066254  152777  000010  124732            BISB    #10,@ML.REG+40                  ;                               8581
18961 066262  012767  177777  123310            MOV     #-1,IO.BUF                      ;                               8582
18962 066270  152777  000002  124776            BISB    #2,@ML.REG+120                  ;                               8583
18963 066276  010201                            MOV     R2,R1                           ; DSA.ADRS,*                    8584
18964 066300  005201                            INC     R1
18965 066302  010177  124676                    MOV     R1,@ML.REG+30
18966 066306  012777  011600  124660            MOV     #IO.BUF,@ML.REG+20              ;                               8585
18967 066314  016702  124602                    MOV     W.C.SIZE,R2                     ;                               8586
18968 066320  062702  000400                    ADD     #400,R2
18969 066324  010277  124634                    MOV     R2,@ML.REG+10
18970 066330  142777  000077  124616            BICB    #77,@ML.REG                     ;                               8587
18971 066336  152777  000051  124610            BISB    #51,@ML.REG
18972 066344  105777  124654          8$:       TSTB    @ML.REG+50
18973 066350  100375                            BPL     8$
18974 066352  032777  040000  124634            BIT     #40000,@ML.REG+40               ;                               8590
18975 066360  001441                            BEQ     9$
18976 066362  104455                            TRAP    55                              ;                               8593
18977 066364  000160                            .WORD   160
18978 066366  010464                            .WORD   ASYNC
18979 066370  024060                            .WORD   DUMPER
18980 066372  012746  006442                    MOV     #WRD.14,-(SP)                   ;                               8594
18981 066376  012746  007156                    MOV     #WRD.60,-(SP)
18982 066402  012746  007046                    MOV     #WRD.50,-(SP)
18983 066406  012746  010162                    MOV     #FNC.17,-(SP)
18984 066412  012746  006046                    MOV     #FOR.FMT,-(SP)
18985 066416  012746  000005                    MOV     #5,-(SP)
18986 066422  010600                            MOV     SP,R0                           ; SP,*
18987 066424  104414                            TRAP    14
18988 066426  017716  124552                    MOV     @ML.REG+30,(SP)                 ;                               8595
18989 066432  005316                            DEC     (SP)
18990 066434  012746  004644                    MOV     #FMT.9,-(SP)
18991 066440  012746  000002                    MOV     #2,-(SP)
18992 066444  010600                            MOV     SP,R0                           ; SP,*
18993 066446  104414                            TRAP    14
18994 066450  016700  125124                    MOV     ML.LUN,R0                       ;                               8596
18995 066454  104451                            TRAP    51
18996 066456  104444                            TRAP    44
18997 066460  062706  000020                    ADD     #20,SP                          ;                               8592
18998 066464  000207                    9$:     RTS     PC                              ;                               8479
18999
19000                                   ; Routine Size:  245 words
19001                                   ; Maximum stack depth per invocation:  14 words
```

```
19010
19011
19015
19016                                   .SBTTL  T40
19020 066466                    T40::
19021 066466  004767  177022      1$:   JSR     PC,$T40                         ;                8600
19022 066472  104466                    TRAP    66
19023 066474  006000                    ROR     R0
19024 066476  103773                    BLO     1$
19025 066500  000207                    RTS     PC
19026
19027                           ; Routine Size:  6 words
19028                           ; Maximum stack depth per invocation:  0 words
19033
19034
19035 ;          8603  !<BLF/PAGE>
```

```
19037 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
19038 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (91)
19039
19040 ;          8604  !
19041 ;          8605  BGNTST;
19042 ;          8606
19043 ;          8607  !++
19044 ;          8608  ! TEST NUMBER:  TST 41
19045 ;          8609  !
19046 ;          8610  ! TEST NAME:    NIBBLE OFFSET TEST
19047 ;          8611  !
19048 ;          8612  ! TEST DESCRIPTION:
19049 ;          8613  !       TEST NIBBLE OFFSET COUNTERS TO OFFSET GOOD NIBBLE DATA A MAX
19050 ;          8614  !       OF 14 WORDS ON DETECTION OF ALL BAD NIBBLES BY:
19051 ;          8615  !
19052 ;          8616  !       1.  LOADING FIRST 2 1/4 WORDS OF THE IO_BUF WITH ZEROES AND THE
19053 ;          8617  !           REMAINING OF BUFFER WITH ONES.
19054 ;          8618  !
19055 ;          8619  !       2.  VIA DAT_DM MODE WRITE THE GOOD BLOCK WITH BACKGROUND
19056 ;          8620  !           ON ONES.
19057 ;          8621  !
19058 ;          8622  !       3.  VIA PROM R/W MODE FORCE ALL ARRAY NIBBLES BAD.
19059 ;          8623  !
19060 ;          8624  !       4.  VIA A MBUS WRITE FUNCTION LOAD IO_BUF INTO THE GOOD BLOCK.
19061 ;          8625  !
19062 ;          8626  !       5.  VIA DAT_DM MODE READ FIRST 15 ARRAY WORDS FOR ZEROES AND THE
19063 ;          8627  !           REMAINING WORDS FOR ONES.
19064 ;          8628  !
19065 ;          8629  ! IMPLICIT INPUTS:
19066 ;          8630  !       PD_TEMP
19067 ;          8631  !       A BIT VECTOR OF 16 BITS WHERE THE READ PROM DATA IS STORED AND
19068 ;          8632  !       ACCESSED FROM.
19069 ;          8633  !
19070 ;          8634  !       IO_BUF
19071 ;          8635  !       A VECTOR OF 256 WORDS WHERE DATA FOR MBUS READS AND WRITE
19072 ;          8636  !       FUNCTIONS ARE FOUND.
19073 ;          8637  !--
19074 ;          8638
19075 ;          8639  local
19076 ;          8640      DODU_FLG,                          !DROP UNIT FLAG
19077 ;          8641      TST_PAT,                           !TEST PATTERN
19078 ;          8642      START,                             !STARTING WORD
19079 ;          8643      FINISH,                            !ENDING WORD
19080 ;          8644      ERR_FLG;                           !ERROR FLAG
19081 ;          8645
19082 ;          8646  BGNSUB;
19083 ;          8647  CLR_MBUS;
19084 ;          8648  DODU_FLG = ZERO;
19085 ;          8649
19086 ;          8650  incr WD_CNT from 0 to 255 do           !LOAD IO_BUF WITH ONES
19087 ;          8651      IO_BUF [.WD_CNT] = ONES;
19088 ;          8652
19089 ;          8653  IO_BUF [0] = ZEROES;                   !LOAD FIRST 2 1/4 WORDS WITH ZEROES
19090 ;          8654  IO_BUF [1] = ZEROES;
19091 ;          8655  IO_BUF [2] = %o'177760';
19092 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
19093 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (91)
```

```
19094
19095 ;          8656  MLD1 = ONES;                                  !LOAD DATA DIAG REGISTERS WITH ONES
19096 ;          8657  MLD2 = ONES;
19097 ;          8658  MLE2 = ONES;
19098 ;          8659  DAT_DM_XFER ();                               !SET UP A DATA DIAG MODE XFERR
19099 ;          8660  MLCS1 = write;                                !DO A WRITE XFERR
19100 ;          8661
19101 ;          8662  incr WD_CNT from 0 to 127 do                  !LOAD BLOCK WITH BG PAT
19102 ;          8663      begin
19103 ;          8664      DELAY (ONE_US);
19104 ;          8665      DAT_CLK = ONE;
19105 ;          8666      end;
19106 ;          8667
19107 ;          8668  CLR_MBUS;
19108 ;          8669  WRT_PD (ONES, 19);                            !FORCE ALL NIBBLES BAD
19109 ;          8670  PROM_RW = ONE;                                !SET PROM READ WRITE
19110 ;          8671  GD_BLK_XFER ();                               .SET UP A GOOD BLOCK XFERR
19111 ;          8672  MLCS1 = write;                                !WRITE ZEROES IN FIRST BLOCK WORD LOCATION
19112 ;          8673                                                !OFFSETTING 14 NIBBLES WITH ZEROES ALSO.
19113 ;          8674  TIME_OUT_LOOP;
19114 ;          8675  CLR_MBUS;
19115 ;          8676  START = ZERO;                                 !START AT THE FIRST BLOCK WORD
19116 ;          8677  FINISH = 14;                                  !END AT THE 14'TH BLOCK WORD
19117 ;          8678  TST_PAT = ZEROES;                             !TEST FOR ZEROES IN FIRST 14 WORDS
19118 ;          8679  DAT_DM_XFER ();                               !SET UP A DATA DIAG MODE XFERR
19119 ;          8680  MLCS1 = read;                                 !DO A READ FUNCTION
19120 ;          8681  DELAY (ONE_US);
19121 ;          8682
19122 ;          8683  incr TWICE from 0 to 1 do                     !READ WORDS 0-14 FOR 0'S AND 15-126 FOR 1'S
19123 ;          8684      begin
19124 ;          8685
19125 ;          8686      incr WRD_CNT from .START to .FINISH do     !READ BLOCK WORDS FORM START TO FINISH
19126 ;          8687          begin
19127 ;          8688          PD_TEMP = .MLPD;                       !GET PROM DATA
19128 ;          8689          DAT_CLK = ONE;                         !CLOCK OUT THE DATA WORD
19129 ;          8690          DELAY (ONE_US);
19130 ;          8691          RD_LNG_WRD;                            !READ THE DATA DIAG REGISTERS
19131 ;          8692
19132 ;          8693          incr NIB_PTR from 0 to 8 do            !LOOK AT 9 NIBBLES
19133 ;          8694
19134 ;          8695              if .PD_TEMP [.NIB_PTR] IS_NOT_SET  !FIND GOOD NIBBLES
19135 ;          8696              then
19136 ;          8697                  begin
19137 ;          8698                  TST_LNG_WRD (.NIB_PTR, .TST_PAT, ERR_FLG);      !COMPARE NIBBLE WITH TST PAT
19138 ;          8699
19139 ;          8700                  if .ERR_FLG IS_SET             !SEE IF COMPARE FOUND AN ERROR
19140 ;          8701                  then
19141 ;          8702                      begin                      !ERROR AND SET DODU_FLG IF ERROR FLG SET
19142 ;          8703                      ERRDF (94, ARR_DAT, DUMPER);
19143 ;          8704                      PRINTB (THR_FMT, WRD_41, WRD_46, WRD_10);
19144 ;          8705                      PRINTB (FMT_6, .NIB_PTR);
19145 ;          8706                      DODU_FLG = ONE;
19146 ;          8707                      end;
19147 ;ML4                                                  14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
19148 ;                                                     14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (91)
19149
19150 ;          8708
```

```
 19151 ;          8709                    end;
 19152 ;          8710
 19153 ;          8711                end;
 19154 ;          8712
 19155 ;          8713            TST_PAT = not .TST_PAT;                 !NOW READ FOR ONES
 19156 ;          8714            START = 15;                             !START A 15
 19157 ;          8715            FINISH = 126;                           !END AT 126
 19158 ;          8716            end
 19159 ;          8717
 19160 ;          8718        ENDSUB;
 19161 ;          8719
 19162 ;          8720        if .DODU_FLG IS_SET                         !DROP THIS UNIT IF DODU_FLG SET
 19163 ;          8721        then
 19164 ;          8722            begin
 19165 ;          8723            DODU (.ML_LUN);
 19166 ;          8724            DOCLN;
 19167 ;          8725            end;
 19168 ;          8726
 19169 ;          8727        ENDTST;
 19173
 19174                                        .SBTTL  $T41
 19178 066502  004167  115410          $T41: JSR     R1,$SAVE5                         ;                       8602
 19179 066506  162706  000012                SUB     #12,SP
 19180 066512  104402                  1$:   TRAP    2                                 ;                       8644
 19181 066514  152777  000040  124472        BISB    #40,@ML.REG+40                    ;                       8646
 19182 066522  016704  125054                MOV     ML.DUT,R4
 19183 066526  042704  177770                BIC     #177770,R4
 19184 066532  142777  000007  124454        BICB    #7,@ML.REG+40
 19185 066540  150477  124450                BISB    R4,@ML.REG+40
 19186 066544  005066  000004                CLR     4(SP)                             ; DODU.FLG              8648
 19187 066550  005002                        CLR     R2                                ; WD.CNT                8650
 19188 066552  010203                  2$:   MOV     R2,R3                             ; WD.CNT,*              8651
 19189 066554  006303                        ASL     R3
 19190 066556  012763  177777  011600        MOV     #-1,IO.BUF(R3)
 19191 066564  005202                        INC     R2                                ; WD.CNT                8650
 19192 066566  020227  000377                CMP     R2,#377                           ; WD.CNT,*
 19193 066572  003767                        BLE     2$
 19194 066574  005067  123000                CLR     IO.BUF                            ;                       8653
 19195 066600  005067  122776                CLR     IO.BUF+2                          ;                       8654
 19196 066604  012767  177760  122772        MOV     #-20,IO.BUF+4                     ;                       8655
 19197 066612  012777  177777  124524        MOV     #-1,@ML.REG+170                   ;                       8656
 19198 066620  012777  177777  124526        MOV     #-1,@ML.REG+200                   ;                       8657
 19199 066626  012777  177777  124500        MOV     #-1,@ML.REG+160                   ;                       8658
 19200 066634  004767  125322                JSR     PC,DAT.DM.XFER                    ;                       8659
 19201 066640  012777  000061  124306        MOV     #61,@ML.REG                       ;                       8660
 19202                               ;ML4                                                14-May-1981 13:48:54    TOPS
 19203                               ;                                                   14-May-1981 13:05:08    PA:<
 19204
 19205 066646  005001                  3$:   CLR     R1                                ; WD.CNT                8662
 19206 066650  012702  000001                MOV     #1,R2                             ; *,$$TMP2              8664
 19207 066654  001411                  4$:   BEQ     7$
 19208 066656  016703  113234                MOV     L$DLY,R3                          ; *,$$TMP1
 19209 066662  001404                        BEQ     6$
 19210 066664  005066  000010          5$:   CLR     10(SP)                            ; $$TMP
 19211 066670  005303                        DEC     R3                                ; $$TMP1
 19212 066672  001374                        BNE     5$
 19213 066674  005302                  6$:   DEC     R2                                ; $$TMP2
```

```
19214 066676  000766                        BR      4$
19215 066700  152777  000020  124366  7$:   BISB    #20,@ML.REG+120     ;                                   8665
19216 066706  005201                        INC     R1                  ; WD.CNT                            8662
19217 066710  020127  000177                CMP     R1,#177             ; WD.CNT,*
19218 066714  003755                        BLE     3$
19219 066716  152777  000040  124270        BISB    #40,@ML.REG+40      ;                                   8666
19220 066724  016704  124652                MOV     ML.DUT,R4
19221 066730  042704  177770                BIC     #177770,R4
19222 066734  142777  000007  124252        BICB    #7,@ML.REG+40
19223 066742  150477  124246                BISB    R4,@ML.REG+40
19224 066746  012746  177777                MOV     #-1,-(SP)           ;                                   8669
19225 066752  012746  000023                MOV     #23,-(SP)
19226 066756  004767  132244                JSR     PC,WRT.PD
19227 066762  152777  000100  124304        BISB    #100,@ML.REG+120    ;                                   8670
19228 066770  004767  127074                JSR     PC,GD.BLK.XFER      ;                                   8671
19229 066774  012777  000061  124152        MOV     #61,@ML.REG         ;                                   8672
19230 067002  105777  124216        8$:     TSTB    @ML.REG+50
19231 067006  100375                        BPL     8$
19232 067010  152777  000040  124176        BISB    #40,@ML.REG+40      ;                                   8674
19233 067016  016704  124560                MOV     ML.DUT,R4
19234 067022  042704  177770                BIC     #177770,R4
19235 067026  142777  000007  124160        BICB    #7,@ML.REG+40
19236 067034  150477  124154                BISB    R4,@ML.REG+40
19237 067040  005066  000006                CLR     6(SP)               ; START                             8676
19238 067044  012766  000016  000004        MOV     #16,4(SP)           ; *,FINISH                          8677
19239 067052  005005                        CLR     R5                  ; TST.PAT                           8678
19240 067054  004767  125102                JSR     PC,DAT.DM.XFER      ;                                   8679
19241 067060  012777  000071  124066        MOV     #71,@ML.REG         ;                                   8680
19242 067066  012702  000001                MOV     #1,R2               ; *,$$TMP2                          8681
19243 067072  001411              9$:       BEQ     12$
19244 067074  016703  113016                MOV     L$DLY,R3            ; *,$$TMP1
19245 067100  001404                        BEQ     11$
19246 067102  005066  000014        10$:    CLR     14(SP)              ; $$TMP
19247 067106  005303                        DEC     R3                  ; $$TMP1
19248 067110  001374                        BNE     10$
19249 067112  005302              11$:      DEC     R2                  ; $$TMP2
19250 067114  000766                        BR      9$
19251 067116  005004              12$:      CLR     R4                  ; TWICE                             8683
19252 067120  016601  000006        13$:    MOV     6(SP),R1            ; START,WRD.CNT                     8686
19253 067124  005301                        DEC     R1                  ; WRD.CNT
19254 067126  000534                        BR      22$
19255 067130  017767  124250  123762  14$:  MOV     @ML.REG+230,PD.TEMP ;                                   8688
19256 067136  152777  000020  124130        BISB    #20,@ML.REG+120     ;                                   8689
19257                             ;ML4                                                14-May-1981 13:48:54  TOPS
19258                             ;                                                   14-May-1981 13:05:08  PA:<
19259
19260 067144  012702  000001                MOV     #1,R2               ; *,$$TMP2                          8690
19261 067150  001411              15$:      BEQ     18$
19262 067152  016703  112740                MOV     L$DLY,R3            ; *,$$TMP1
19263 067156  001404                        BEQ     17$
19264 067160  005066  000014        16$:    CLR     14(SP)              ; $$TMP
19265 067164  005303                        DEC     R3                  ; $$TMP1
19266 067166  001374                        BNE     16$
19267 067170  005302              17$:      DEC     R2                  ; $$TMP2
19268 067172  000766                        BR      15$
19269 067174  017767  124144  121750  18$:  MOV     @ML.REG+170,D1.TEMP
19270 067202  017767  124146  121744        MOV     @ML.REG+200,D2.TEMP
```

```
19271 067210  017767  124120  121740          MOV     @ML.REG+160,E2.TEMP
19272 067216  005002                          CLR     R2                      ; NIB.PTR                       8693
19273 067220  010203                  19$:    MOV     R2,R3                   ; NIB.PTR,*                     8695
19274 067222  006203                          ASR     R3
19275 067224  006203                          ASR     R3
19276 067226  006203                          ASR     R3
19277 067230  062703  013120                  ADD     #PD.TEMP,R3
19278 067234  010346                          MOV     R3,-(SP)
19279 067236  010246                          MOV     R2,-(SP)                ; NIB.PTR,*
19280 067240  042716  177770                  BIC     #177770,(SP)
19281 067244  012746  000001                  MOV     #1,-(SP)
19282 067250  005046                          CLR     -(SP)
19283 067252  004767  113662                  JSR     PC,BL$GT2
19284 067256  062706  000010                  ADD     #10,SP
19285 067262  005700                          TST     R0
19286 067264  001051                          BNE     21$
19287 067266  010246                          MOV     R2,-(SP)                ; NIB.PTR,*                     8698
19288 067270  010546                          MOV     R5,-(SP)                ; TST.PAT,*
19289 067272  012746  000020                  MOV     #20,-(SP)
19290 067276  060616                          ADD     SP,(SP)                 ; ERR.FLG,*
19291 067300  004767  126650                  JSR     PC,TST.LNG.WRD
19292 067304  026627  000020  000001          CMP     20(SP),#1               ; ERR.FLG,*                     8700
19293 067312  001034                          BNE     20$
19294 067314  104455                          TRAP    55                      ;                               8703
19295 067316  000136                          .WORD   136
19296 067320  010570                          .WORD   ARR.DAT
19297 067322  024060                          .WORD   DUMPER
19298 067324  012746  006406                  MOV     #WRD.10,-(SP)           ;                               8704
19299 067330  012746  007004                  MOV     #WRD.46,-(SP)
19300 067334  012746  006732                  MOV     #WRD.41,-(SP)
19301 067340  012746  006034                  MOV     #THR.FMT,-(SP)
19302 067344  012746  000004                  MOV     #4,-(SP)
19303 067350  010600                          MOV     SP,R0                   ; SP,*
19304 067352  104414                          TRAP    14
19305 067354  010216                          MOV     R2,(SP)                 ; NIB.PTR,*                     8705
19306 067356  012746  004532                  MOV     #FMT.6,-(SP)
19307 067362  012746  000002                  MOV     #2,-(SP)
19308 067366  010600                          MOV     SP,R0                   ; SP,*
19309 067370  104414                          TRAP    14
19310 067372  012766  000001  000034          MOV     #1,34(SP)               ; *,DODU.FLG                    8706
19311 067400  062706  000016                  ADD     #16,SP                  ;                               8702
19312                               ;ML4                                                14-May-1981 13:48:54    TOPS
19313                               ;                                                   14-May-1981 13:05:08    PA:<
19314
19315 067404  062706  000006          20$:    ADD     #6,SP                   ;                               8697
19316 067410  005202                  21$:    INC     R2                      ; NIB.PTR                       8693
19317 067412  020227  000010                  CMP     R2,#10                  ; NIB.PTR,*
19318 067416  003700                          BLE     19$
19319 067420  005201                  22$:    INC     R1                      ; WRD.CNT                       8686
19320 067422  020166  000004                  CMP     R1,4(SP)                ; WRD.CNT,FINISH
19321 067426  003640                          BLE     14$
19322 067430  005105                          COM     R5                      ; TST.PAT                       8713
19323 067432  012766  000017  000006          MOV     #17,6(SP)               ; *,START                       8714
19324 067440  012766  000176  000004          MOV     #176,4(SP)              ; *,FINISH                      8715
19325 067446  005204                          INC     R4                      ; TWICE                         8683
19326 067450  020427  000001                  CMP     R4,#1                   ; TWICE,*
19327 067454  003621                          BLE     13$
```

```
 19328 067456  022626                        CMP     (SP)+,(SP)+                     ;                                  8644
 19329 067460  104467                        TRAP    67                              ;                                  8716
 19330 067462  006000                        ROR     R0
 19331 067464  103002                        BHIS    23$
 19332 067466  000167  177020               JMP     1$
 19333 067472  026627  000004  000001  23$:  CMP     4(SP),#1                        ; DODU.FLG,*                       8720
 19334 067500  001004                        BNE     24$
 19335 067502  016700  124072               MOV     ML.LUN,R0                       ;                                  8723
 19336 067506  104451                        TRAP    51
 19337 067510  104444                        TRAP    44
 19338 067512  062706  000012        24$:    ADD     #12,SP                          ;                                  8602
 19339 067516  000207                        RTS     PC
 19340
 19341                               ; Routine Size:  263 words
 19342                               ; Maximum stack depth per invocation:  23 words
 19347
 19348
 19352
 19353                                       .SBTTL  T41
 19357 067520                        T41::
 19358 067520  004767  176756        1$:     JSR     PC,$T41                         ;                                  8725
 19359 067524  104466                        TRAP    66
 19360 067526  006000                        ROR     R0
 19361 067530  103773                        BLO     1$
 19362 067532  000207                        RTS     PC
 19363
 19364                               ; Routine Size:  6 words
 19365                               ; Maximum stack depth per invocation:  0 words
 19373
 19374
 19375 ;          8728  !<BLF/PAGE>
```

```
19377 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
19378 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (92)
19379
19380 ;          8729  !
19381 ;          8730  BGNTST;
19382 ;          8731
19383 ;          8732  !++
19384 ;          8733  ! TEST NUMBER:  TST 42
19385 ;          8734  !
19386 ;          8735  ! TEST NAME:    CS1 FUNCTION ABORT TEST
19387 ;          8736  !
19388 ;          8737  ! TEST DESCRIPTION:
19389 ;          8738  !       TEST CS1 FUNCTION ABORTS ON DETECTION OF CLASS 'A' & 'B' ERRORS BY:
19390 ;          8739  !
19391 ;          8740  !       1.  VIA MBUS WRITE FUNCTION LOAD THE GOOD BLOCK WITH BACKGROUND
19392 ;          8741  !           PATTERN OF ONES.
19393 ;          8742  !
19394 ;          8743  !       2.  CLEAR THE IO_BUF
19395 ;          8744  !
19396 ;          8745  !       3.  DO A MBUS READ FUNCTION.  WHILE THE READ IS IN PROGRESS WRITE TO
19397 ;          8746  !           MLDA (CLASS 'A' ERROR) READ THE IO_BUF FOR ONES.
19398 ;          8747  !
19399 ;          8748  !       4.  CLEAR THE IO_BUF
19400 ;          8749  !
19401 ;          8750  !       5.  VIA PROM R/W MODE FORCE UV ERROR TO THE UV ADRS ERROR PROM
19402 ;          8751  !           (CLASS 'B' ERROR).  DO A MBUS READ.
19403 ;          8752  !
19404 ;          8753  !       6.  READ IO_BUF FOR ZEROES
19405 ;          8754  !
19406 ;          8755  ! IMPLICIT INPUTS:
19407 ;          8756  !       IO_BUF
19408 ;          8757  !       A VECTOR OF 256 WORDS WHERE DATA FOR MBUS READS AND
19409 ;          8758  !       WRITE FUNCTION ARE FOUND.
19410 ;          8759  !--
19411 ;          8760
19412 ;          8761  CLR_MBUS;
19413 ;          8762  BAI = ONE;                                  !SET ON FIRST IO_BUF ADRS
19414 ;          8763  IO_BUF = ONES;                              !LOAD FIRST IO_BUF ADRS
19415 ;          8764  GD_BLK_XFER ();                             !SET UP A GOOD BLOCK XFERR
19416 ;          8765  MLCS1 = write;                              !WRITE BACKGROUND PATTERN
19417 ;          8766  TIME_OUT_LOOP;
19418 ;          8767
19419 ;          8768  incr TWICE from 0 to 1 do                   !FORCE CLASS 'A' AND CLASS 'B' ERRORS
19420 ;          8769      begin
19421 ;          8770      BGNSUB;
19422 ;          8771      CLR_MBUS;
19423 ;          8772
19424 ;          8773      incr CNT from 0 to 255 do
19425 ;          8774          IO_BUF [.CNT] = ZEROES;             !CLEAR OUT IO_BUF
19426 ;          8775
19427 ;          8776      if .TWICE eql 1                         !IF 2ND PASS THEN FORCE 'B' ERROR
19428 ;          8777      then
19429 ;          8778          begin
19430 ;          8779          PROM_RW = ONE;
19431 ;          8780          MLPD = %o'777';
```

```
19433 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
19434 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (92)
19435
19436 ;          8781          end;
19437 ;          8782
19438 ;          8783      GD_BLK_XFER ();                    !SET UP A GOOD BLOCK XFERR
19439 ;          8784      MLCS1 = read;                      !DO A READ FUNCTION
19440 ;          8785
19441 ;          8786      if .TWICE eql 0 then MLDA = ONES;  !IF FIRST PASS THEN FORCE AN 'A' ERROR
19442 ;          8787
19443 ;          8788      TIME_OUT_LOOP;
19444 ;          8789
19445 ;          8790      if .TWICE eql 0                    !SEE WHICH PASS WE'RE ON
19446 ;          8791      then
19447 ;          8792          begin                          !CLASS 'A' ERROR
19448 ;          8793
19449 ;          8794          incr WRD_CNT from 0 to 64 do   !SEE IF XFERR WAY ALLOWED TO CONTINUE
19450 ;          8795              begin
19451 ;          8796
19452 ;          8797              if .IO_BUF [.WRD_CNT] neq ONES   !READ IO_BUF FOR BG PAT
19453 ;          8798              then
19454 ;          8799                  begin                    !ERROR AND EXIT LOOP IF ZEROES
19455 ;          8800                  ERRDF (95, SYNC, DUMPER);
19456 ;          8801                  PRINTB (FOR_FMT, PHR_8, FNC_13, WRD_19, WRD_10);
19457 ;          8802                  exitloop;
19458 ;          8803                  end;
19459 ;          8804
19460 ;          8805              end;
19461 ;          8806
19462 ;          8807          end
19463 ;          8808      else                               !CLASS 'B' ERROR
19464 ;          8809          begin
19465 ;          8810
19466 ;          8811          incr WRD_CNT from 0 to 64 do   !SEE IF XFERR WAS ABORTED
19467 ;          8812              begin
19468 ;          8813
19469 ;          8814              if .IO_BUF [.WRD_CNT] neq ZEROES   !READ IO_BUF CLEARED DATA
19470 ;          8815              then
19471 ;          8816                  begin                    !ERROR IF ONES AND EXIT LOOP
19472 ;          8817                  ERRDF (96, SYNC, DUMPER);
19473 ;          8818                  PRINTB (FOR_FMT, PHR_9, FNC_13, WRD_19, WRD_10);
19474 ;          8819                  exitloop
19475 ;          8820                  end;
19476 ;          8821
19477 ;          8822              end;
19478 ;          8823
19479 ;          8824          end;
19480 ;          8825
19481 ;          8826      if .SC IS_NOT_SET                  !SEE IF SC BIT SET
19482 ;          8827      then
19483 ;          8828          begin                          !ERROR IF NOT SET
19484 ;          8829          ERRDF (104, SYNC, DUMPER);
19485 ;          8830          PRINTB (FIV_FMT, WRD_59, PHR_1, WRD_11, WRD_19, FNC_13)
19486 ;          8831          end;
19487 ;          8832
```

```
19489 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
19490 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (92)
19491
19492 ;          8833      ENDSUB;
19493 ;          8834      end;
19494 ;          8835
19495 ;          8836  ENDTST;
19499
19500                                        .SBTTL  $T42
19504 067534  004167  114320          $T42: JSR     R1,$SAVE3                 ;                              8727
19505 067540  152777  000040  123446        BISB    #40,@ML.REG+40            ;                              8730
19506 067546  016703  124030                MOV     ML.DUT,R3
19507 067552  042703  177770                BIC     #177770,R3
19508 067556  142777  000007  123430        BICB    #7,@ML.REG+40
19509 067564  150377  123424                BISB    R3,@ML.REG+40
19510 067570  152777  000010  123416        BISB    #10,@ML.REG+40            ;                              8762
19511 067576  012767  177777  121774        MOV     #-1,IO.BUF                ;                              8763
19512 067604  004767  126260                JSR     PC,GD.BLK.XFER            ;                              8764
19513 067610  012777  000061  123336        MOV     #61,@ML.REG               ;                              8765
19514 067616  105777  123402          1$:   TSTB    @ML.REG+50
19515 067622  100375                BPL     1$
19516 067624  005003                CLR     R3                                ; TWICE                        8768
19517 067626  104402                TRAP    2                                 ;                              8769
19518 067630  152777  000040  123356        BISB    #40,@ML.REG+40            ;                              8770
19519 067636  016702  123740                MOV     ML.DUT,R2
19520 067642  042702  177770                BIC     #177770,R2
19521 067646  142777  000007  123340        BICB    #7,@ML.REG+40
19522 067654  150277  123334                BISB    R2,@ML.REG+40
19523 067660  005000                CLR     R0                                ; CNT                          8773
19524 067662  010001          3$:   MOV     R0,R1                             ; CNT,*                        8774
19525 067664  006301                ASL     R1
19526 067666  005061  011600              CLR     IO.BUF(R1)
19527 067672  005200                INC     R0                                ; CNT                          8773
19528 067674  020027  000377              CMP     R0,#377                   ; CNT,*
19529 067700  003770                BLE     3$
19530 067702  020327  000001              CMP     R3,#1                     ; TWICE,*                      8776
19531 067706  001006                BNE     4$
19532 067710  152777  000100  123356        BISB    #100,@ML.REG+120          ;                              8779
19533 067716  012777  000777  123460        MOV     #777,@ML.REG+230          ;                              8780
19534 067724  004767  126140          4$:   JSR     PC,GD.BLK.XFER            ;                              8783
19535 067730  012777  000071  123216        MOV     #71,@ML.REG               ;                              8784
19536 067736  005002                CLR     R2                                ;                              8786
19537 067740  005703                TST     R3                                ; TWICE
19538 067742  001004                BNE     5$
19539 067744  005202                INC     R2
19540 067746  012777  177777  123230        MOV     #-1,@ML.REG+30
19541 067754  105777  123244          5$:   TSTB    @ML.REG+50
19542 067760  100375                BPL     5$
19543 067762  006002                ROR     R2                                ;                              8790
```

```
19545                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
19546                                   ;                                                       14-May-1981 13:05:08    PA:<
19547
19548 067764  103037                            BCC     8$
19549 067766  005002                            CLR     R2                      ; WRD.CNT                               8794
19550 067770  010201                    6$:     MOV     R2,R1                   ; WRD.CNT,*                             8797
19551 067772  006301                            ASL     R1
19552 067774  026127  011600  177777            CMP     IO.BUF(R1),#-1
19553 070002  001423                            BEQ     7$
19554 070004  104455                            TRAP    55                      ;                                       8800
19555 070006  000137                            .WORD   137
19556 070010  010526                            .WORD   SYNC
19557 070012  024060                            .WORD   DUMPER
19558 070014  012746  006406                    MOV     #WRD.10,-(SP)           ;                                       8801
19559 070020  012746  006506                    MOV     #WRD.19,-(SP)
19560 070024  012746  010112                    MOV     #FNC.13,-(SP)
19561 070030  012746  007540                    MOV     #PHR.8,-(SP)
19562 070034  012746  006046                    MOV     #FOR.FMT,-(SP)
19563 070040  012746  000005                    MOV     #5,-(SP)
19564 070044  010600                            MOV     SP,R0                   ; SP,*
19565 070046  104414                            TRAP    14
19566 070050  000435                            BR      10$                     ;                                       8802
19567 070052  005202                    7$:     INC     R2                      ; WRD.CNT                               8794
19568 070054  020227  000100                    CMP     R2,#100                 ; WRD.CNT,*
19569 070060  003743                            BLE     6$
19570 070062  000437                            BR      12$                     ;                                       8790
19571 070064  005002                    8$:     CLR     R2                      ; WRD.CNT                               8811
19572 070066  010201                    9$:     MOV     R2,R1                   ; WRD.CNT,*                             8814
19573 070070  006301                            ASL     R1
19574 070072  005761  011600                    TST     IO.BUF(R1)
19575 070076  001425                            BEQ     11$
19576 070100  104455                            TRAP    55                      ;                                       8817
19577 070102  000140                            .WORD   140
19578 070104  010526                            .WORD   SYNC
19579 070106  024060                            .WORD   DUMPER
19580 070110  012746  006406                    MOV     #WRD.10,-(SP)           ;                                       8818
19581 070114  012746  006506                    MOV     #WRD.19,-(SP)
19582 070120  012746  010112                    MOV     #FNC.13,-(SP)
19583 070124  012746  007552                    MOV     #PHR.9,-(SP)
19584 070130  012746  006046                    MOV     #FOR.FMT,-(SP)
19585 070134  012746  000005                    MOV     #5,-(SP)
19586 070140  010600                            MOV     SP,R0                   ; SP,*
19587 070142  104414                            TRAP    14
19588 070144  062706  000014            10$:    ADD     #14,SP                  ;                                       8819
19589 070150  000404                            BR      12$
19590 070152  005202                    11$:    INC     R2                      ; WRD.CNT                               8811
19591 070154  020227  000100                    CMP     R2,#100                 ; WRD.CNT,*
19592 070160  003742                            BLE     9$
19593 070162  032777  100000  122764    12$:    BIT     #100000,@ML.REG         ;                                       8826
19594 070170  001026                            BNE     13$
19595 070172  104455                            TRAP    55                      ;                                       8829
19596 070174  000150                            .WORD   150
19597 070176  010526                            .WORD   SYNC
19598 070200  024060                            .WORD   DUMPER
19599 070202  012746  010112                    MOV     #FNC.13,-(SP)           ;                                       8830
```

```
19601                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
19602                                   ;                                                       14-May-1981 13:05:08    PA:<
19603
19604 070206  012746  006506                      MOV     #WRD.19,-(SP)
19605 070212  012746  006416                      MOV     #WRD.11,-(SP)
19606 070216  012746  007366                      MOV     #PHR.1,-(SP)
19607 070222  012746  007152                      MOV     #WRD.59,-(SP)
19608 070226  012746  006062                      MOV     #FIV.FMT,-(SP)
19609 070232  012746  000006                      MOV     #6,-(SP)
19610 070236  010600                            MOV     SP,R0                           ; SP,*
19611 070240  104414                            TRAP    14
19612 070242  062706  000016                    ADD     #16,SP                          ;                       8828
19613 070246  104467                  13$:      TRAP    67                              ;                       8831
19614 070250  006000                            ROR     R0
19615 070252  103002                            BHIS    15$
19616 070254  000167  177346          14$:      JMP     2$
19617 070260  005203                  15$:      INC     R3                              ; TWICE                 8768
19618 070262  020327  000001                    CMP     R3,#1                           ; TWICE,*
19619 070266  003772                            BLE     14$
19620 070270  000207                            RTS     PC                              ;                       8727
19621
19622                                   ; Routine Size:  175 words
19623                                   ; Maximum stack depth per invocation:  11 words
19628
19629
19633
19634                                           .SBTTL  T42
19638 070272                            T42::
19639 070272  004767  177236            1$:     JSR     PC,$T42                         ;                       8834
19640 070276  104466                            TRAP    66
19641 070300  006000                            ROR     R0
19642 070302  103773                            BLO     1$
19643 070304  000207                            RTS     PC
19644
19645                                   ; Routine Size:  6 words
19646                                   ; Maximum stack depth per invocation:  0 words
19651
19652
19653 ;           8837  !<BLF/PAGE>
```

```
19655 ;ML4
19656 ;
19657
19658 ;          8838  !
19659 ;          8839  !
19660 ;          8840  BGNTST;
19661 ;          8841
19662 ;          8842  !++
19663 ;          8843  ! TEST NUMBER:  TST 43
19664 ;          8844  !
19665 ;          8845  ! TEST NAME:    LAST BLOCK TRANSFER TEST
19666 ;          8846  !
19667 ;          8847  ! TEST DESCRIPTION:
19668 ;          8848  !
19669 ;          8849  !       TEST THE LAST BLOCK INDICATOR BIT
19670 ;          8850  !       FOR SETTING/NOT SETTING AND THE
19671 ;          8851  !       DSA REGISTER FOR INCREMENTING BY:
19672 ;          8852  !
19673 ;          8853  !       1.  DOING MBUS TRANSFERRS AT EACH
19674 ;          8854  !           BLOCK FROM BLOCK ZERO TO
19675 ;          8855  !           LAST BLOCK -1 AND TEST LBT
19676 ;          8856  !           CLEAR AND DSA REGISTER TO BE
19677 ;          8857  !           INCREMENTED.
19678 ;          8858  !
19679 ;          8859  !       2.  DO A MBUS TRANSFERR AT
19680 ;          8860  !           THE LAST BLOCK.
19681 ;          8861  !           TEST LBT TO BE SET AND
19682 ;          8862  !           TEST FOR CLEAR.
19683 ;          8863  !           TEST DSA REG TO BE INCREMENTED.
19684 ;          8864  !
19685 ;          8865  ! IMPLICIT INPUTS:
19686 ;          8866  !       IO_BUF
19687 ;          8867  !       A VECTOR OF 256 WORDS WHERE
19688 ;          8868  !       DATA FOR MBUS READS AND WRITE
19689 ;          8869  !       FUNCTION ARE FOUND.
19690 ;          8870  !
19691 ;          8871  !--
19692 ;          8872
19693 ;          8873  local
19694 ;          8874      DODU_FLG;                           !DROP UNIT FLAG
19695 ;          8875
19696 ;          8876  DODU_FLG = ZERO;
19697 ;          8877
19698 ;          8878  incr DSA_CNT from 0 to .LST_BLK - 1 do      !DO XFERRS UP TO THE LAST BLOCK
19699 ;          8879      begin
19700 ;          8880      BGNSUB;
19701 ;          8881      CLR_MBUS;
19702 ;          8882      ECC_DIS = ONE;                      !DISABLE ECC
19703 ;          8883      MLWC = not 255;                     !LOAD WORD COUNT
19704 ;          8884      MLBA = IO_BUF;                      !LOAD UBUS ADRS
19705 ;          8885      MLDA = .DSA_CNT;                    !LOAD DSA WITH DSA_CNT
19706 ;          8886      MLCS1 = write;                      !DO A WRITE FUNCTION
19707 ;          8887      TIME_OUT_LOOP;
19708 ;          8888
19709 ;          8889      if .LBT IS_SET                      !SEE IF THE LAST BLOCK XFERR BIT SET
```

```
:ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
:                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (93)

:          8890       then
:          8891           begin                                    !ERROR AND SET DODU_FLG IF SET
:          8892           ERRDF (97, ASYNC, DUMPER);
:          8893           PRINTB (THR_FMT, WRD_27, PHR_5, WRD_29);
:          8894           PRINTB (FMT_7, .DSA_CNT);
:          8895           DODU_FLG = ONE;
:          8896           end;
:          8897
:          8898       if .MLDA neq .DSA_CNT + 1                    !SEE IF THE DSA REG INCREMENTED
:          8899       then
:          8900           begin                                    !ERROR AND SET DODU_FLG IF NOT
:          8901           ERRDF (98, ASYNC, DUMPER);
:          8902           PRINTB (THR_FMT, REG_6, WRD_31, WRD_14);
:          8903           PRINTB (FMT_7, .DSA_CNT);
:          8904           DODU_FLG = ONE;
:          8905           end;
:          8906
:          8907       ENDSUB;
:          8908       end;
:          8909
:          8910   BGNSUB;
:          8911   CLR_MBUS;
:          8912   ECC_DIS = ONE;                                   !DISABLE ECC
:          8913   LAST_BLK_XFER ();                                !SET UP A LAST BLOCK XFERR
:          8914   MLCST = write;                                   !DO A WRITE FUNCTION
:          8915   TIME_OUT_LOOP;
:          8916
:          8917   if .MLDA neq .LST_BLK + 1                        !SEE IF DSA REGISTER INCREMENTED
:          8918   then
:          8919       begin                                        !ERROR AND SET DODU_FLG IF NOT
:          8920       ERRDF (101, ASYNC, DUMPER);
:          8921       PRINTB (THR_FMT, REG_6, WRD_31, WRD_14);
:          8922       PRINTB (FMT_7, .LST_BLK);
:          8923       DODU_FLG = ONE;
:          8924       end;
:          8925
:          8926   if .LBT IS_SET                                   !SEE IF LBT BIT SET
:          8927   then
:          8928       begin
:          8929       MLDA = ONES;                                 !IF SET THEN TRY TO CLEAR IT
:          8930
:          8931       if .LBT IS_SET                               !SEE IF BIT CLEARED
:          8932       then
:          8933           begin                                    !ERROR IF NOT
:          8934           ERRDF (99, ASYNC, DUMPER);
:          8935           PRINTB (FIV_FMT, WRD_27, PHR_2, WRD_11, WRD_17, REG_6);
:          8936           DODU_FLG = ONE;
:          8937           end;
:          8938
:          8939       end
:          8940   else                                             !LBT NOT SET
:          8941       begin                                        !ERROR AND SET DODU_FLG
```

```
19767 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
19768 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (93)
19769
19770 ;          8942      ERRDF (100, ASYNC, DUMPER);
19771 ;          8943      PRINTB (FOR_FMT, WRD_27, PHR_1, WRD_11, WRD_27);
19772 ;          8944      DODU_FLG = ONE;
19773 ;          8945      end;
19774 ;          8946
19775 ;          8947  ENDSUB;
19776 ;          8948
19777 ;          8949  if .DODU_FLG IS_SET                     !DROP THIS UNIT IF DODU_FLG SET
19778 ;          8950  then
19779 ;          8951      begin
19780 ;          8952      DODU (.ML_LUN);
19781 ;          8953      DOCLN;
19782 ;          8954      end;
19783 ;          8955
19784 ;          8956  ENDTST;
19788
19789                                        .SBTTL  $T43
19793 070306  004167  113564          $T43:  JSR     R1,$SAVE4                ;                          8836
19794 070312  005001                        CLR     R1                       ; DODU.FLG                 8876
19795 070314  016704  121252                MOV     LST.BLK,R4               ;                          8878
19796 070320  005002                        CLR     R2                       ; DSA.CNT
19797 070322  000541                        BR      6$
19798 070324  010203                  1$:   MOV     R2,R3                    ; DSA.CNT,*                8898
19799 070326  005203                        INC     R3
19800 070330  104402                  2$:   TRAP    2                        ;                          8879
19801 070332  152777  000040  122654        BISB    #40,@ML.REG+40           ;                          8880
19802 070340  016700  123236                MOV     ML.DUT,R0
19803 070344  042700  177770                BIC     #177770,R0
19804 070350  142777  000007  122636        BICB    #7,@ML.REG+40
19805 070356  150077  122632                BISB    R0,@ML.REG+40
19806 070362  152777  000002  122704        BISB    #2,@ML.REG+120           ;                          8882
19807 070370  012777  177400  122566        MOV     #-400,@ML.REG+10         ;                          8883
19808 070376  012777  011600  122570        MOV     #IO.BUF,@ML.REG+20       ;                          8884
19809 070404  010277  122574                MOV     R2,@ML.REG+30            ; DSA.CNT,*                8885
19810 070410  012777  000061  122536        MOV     #61,@ML.REG              ;                          8886
19811 070416  105777  122602          3$:   TSTB    @ML.REG+50
19812 070422  100375                        BPL     3$
19813 070424  032777  002000  122572        BIT     #2000,@ML.REG+50         ;                          8889
19814 070432  001433                        BEQ     4$
19815 070434  104455                        TRAP    55                       ;                          8892
19816 070436  000141                        .WORD   141
19817 070440  010464                        .WORD   ASYNC
19818 070442  024060                        .WORD   DUMPER
19819 070444  012746  006610                MOV     #WRD.29,-(SP)            ;                          8893
19820 070450  012746  007472                MOV     #PHR.5,-(SP)
19821 070454  012746  006602                MOV     #WRD.27,-(SP)
```

```
                                                                                                14-May-1981 13:48:54    TOPS
                                        ;ML4                                                    14-May-1981 13:05:08    PA:<
                                        ;
                                                MOV     #THR.FMT,-(SP)
                                                MOV     #4,-(SP)
                                                MOV     SP,R0                   ; SP,*
                                                TRAP    14
                                                MOV     R2,(SP)                 ; DSA.CNT,*                             8894
                                                MOV     #FMT.7,-(SP)
                                                MOV     #2,-(SP)
                                                MOV     SP,R0                   ; SP,*
                                                TRAP    14
                                                MOV     #1,R1                   ; *,DODU.FLG                            8895
                                                ADD     #16,SP                  ;                                       8891
                                        4$:     CMP     @ML.REG+30,R3           ;                                       8898
                                                BEQ     5$
                                                TRAP    55                      ;                                       8901
                                                .WORD   142
                                                .WORD   ASYNC
                                                .WORD   DUMPER
                                                MOV     #WRD.14,-(SP)           ;                                       8902
                                                MOV     #WRD.31,-(SP)
                                                MOV     #REG.6,-(SP)
                                                MOV     #THR.FMT,-(SP)
                                                MOV     #4,-(SP)
                                                MOV     SP,R0                   ; SP,*
                                                TRAP    14
                                                MOV     R2,(SP)                 ; DSA.CNT,*                             8903
                                                MOV     #FMT.7,-(SP)
                                                MOV     #2,-(SP)
                                                MOV     SP,R0                   ; SP,*
                                                TRAP    14
                                                MOV     #1,R1                   ; *,DODU.FLG                            8904
                                                ADD     #16,SP                  ;                                       8900
                                        5$:     TRAP    67                      ;                                       8905
                                                ROR     R0
                                                BLO     2$
                                                INC     R2                      ; DSA.CNT                               8878
                                        6$:     CMP     R2,R4                   ; DSA.CNT,*
                                                BLT     1$
                                        7$:     TRAP    2                       ;                                       8908
                                                BISB    #40,@ML.REG+40          ;                                       8910
                                                MOV     ML.OUT,R4
                                                BIC     #177770,R4
                                                BICB    #7,@ML.REG+40
                                                BISB    R4,@ML.REG+40
                                                BISB    #2,@ML.REG+120          ;                                       8912
                                                JSR     PC,LAST.BLK.XFER        ;                                       8913
                                                MOV     #61,@ML.REG             ;                                       8914
                                        8$:     TSTB    @ML.REG+50
                                                BPL     8$
                                                MOV     LST.BLK,R2              ;                                       8917
                                                INC     R2
                                                CMP     @ML.REG+30,R2
                                                BEQ     9$
```

```
19879                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
19880                                   ;                                                       14-May-1981 13:05:08    PA:<
19881
19882 070726  104455                          TRAP    55                      ;                                       8920
19883 070730  000145                          .WORD   145
19884 070732  010464                          .WORD   ASYNC
19885 070734  024060                          .WORD   DUMPER
19886 070736  012746  006442                  MOV     #WRD.14,-(SP)           ;                                       8921
19887 070742  012746  006632                  MOV     #WRD.31,-(SP)
19888 070746  012746  010316                  MOV     #REG.6,-(SP)
19889 070752  012746  006034                  MOV     #THR.FMT,-(SP)
19890 070756  012746  000004                  MOV     #4,-(SP)
19891 070762  010600                          MOV     SP,R0                   ; SP,*
19892 070764  104414                          TRAP    14
19893 070766  016716  120600                  MOV     LST.BLK,(SP)            ;                                       8922
19894 070772  012746  004562                  MOV     #FMT.7,-(SP)
19895 070776  012746  000002                  MOV     #2,-(SP)
19896 071002  010600                          MOV     SP,R0                   ; SP,*
19897 071004  104414                          TRAP    14
19898 071006  012701  000001                  MOV     #1,R1                   ; *,DODU.FLG                            8923
19899 071012  062706  000016                  ADD     #16,SP                  ;                                       8919
19900 071016  032777  002000  122200  9$:     BIT     #2000,@ML.REG+50        ;                                       8926
19901 071024  001440                          BEQ     10$
19902 071026  012777  177777  122150          MOV     #-1,@ML.REG+30          ;                                       8929
19903 071034  032777  002000  122162          BIT     #2000,@ML.REG+50        ;                                       8931
19904 071042  001457                          BEQ     11$
19905 071044  104455                          TRAP    55                      ;                                       8934
19906 071046  000143                          .WORD   143
19907 071050  010464                          .WORD   ASYNC
19908 071052  024060                          .WORD   DUMPER
19909 071054  012746  010316                  MOV     #REG.6,-(SP)            ;                                       8935
19910 071060  012746  006470                  MOV     #WRD.17,-(SP)
19911 071064  012746  006416                  MOV     #WRD.11,-(SP)
19912 071070  012746  007404                  MOV     #PHR.2,-(SP)
19913 071074  012746  006602                  MOV     #WRD.27,-(SP)
19914 071100  012746  006062                  MOV     #FIV.FMT,-(SP)
19915 071104  012746  000006                  MOV     #6,-(SP)
19916 071110  010600                          MOV     SP,R0                   ; SP,*
19917 071112  104414                          TRAP    14
19918 071114  012701  000001                  MOV     #1,R1                   ; *,DODU.FLG                            8936
19919 071120  062706  000016                  ADD     #16,SP                  ;                                       8933
19920 071124  000426                          BR      11$                     ;                                       8926
19921 071126  104455                10$:      TRAP    55                      ;                                       8942
19922 071130  000144                          .WORD   144
19923 071132  010464                          .WORD   ASYNC
19924 071134  024060                          .WORD   DUMPER
19925 071136  012746  006602                  MOV     #WRD.27,-(SP)           ;                                       8943
19926 071142  012746  006416                  MOV     #WRD.11,-(SP)
19927 071146  012746  007366                  MOV     #PHR.1,-(SP)
19928 071152  012746  006602                  MOV     #WRD.27,-(SP)
19929 071156  012746  006046                  MOV     #FOR.FMT,-(SP)
19930 071162  012746  000005                  MOV     #5,-(SP)
19931 071166  010600                          MOV     SP,R0                   ; SP,*
19932 071170  104414                          TRAP    14
19933 071172  012701  000001                  MOV     #1,R1                   ; *,DODU.FLG                            8944
```

```
                                ;ML4                                                    14-May-1981 13:48:54    TOPS
                                ;                                                       14-May-1981 13:05:08    PA:<
                                
071176  062706  000014                  ADD     #14,SP                          ;                               8941
071202  104467                  11$:    TRAP    67                              ;                               8945
071204  006000                          ROR     R0
071206  103611                          BLO     7$
071210  005301                          DEC     R1                              ; DODU.FLG                      8949
071212  001004                          BNE     12$
071214  016700  122360                  MOV     ML.LUN,R0                       ;                               8952
071220  104451                          TRAP    51
071222  104444                          TRAP    44
071224  000207                  12$:    RTS     PC                              ;                               8836

                                ; Routine Size:  232 words
                                ; Maximum stack depth per invocation:  12 words


                                        .SBTTL  T43
071226                          T43::
071226  004767  177054          1$:     JSR     PC,$T43                         ;                               8954
071232  104466                          TRAP    66
071234  006000                          ROR     R0
071236  103773                          BLO     1$
071240  000207                          RTS     PC

                                ; Routine Size:  6 words
                                ; Maximum stack depth per invocation:  0 words


;       8957    !<BLF/PAGE>
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (94)

```
19982 ;ML4
19983 ;
19984
19985 ;          8958  !
19986 ;          8959  !
19987 ;          8960  BGNTST;
19988 ;          8961
19989 ;          8962  !++
19990 ;          8963  ! TEST NUMBER:  TST 44
19991 ;          8964  !
19992 ;          8965  ! TEST NAME:    INVALID ADRS TEST
19993 ;          8966  !
19994 ;          8967  ! TEST DESCRIPTION:
19995 ;          8968  !
19996 ;          8969  !       TEST THE DETECTION OF ILLEGAL DSA
19997 ;          8970  !       ADDRESSES BY:
19998 ;          8971  !
19999 ;          8972  !       1.  DOING A MBUS WRITE FUNCTION
20000 ;          8973  !           AT ALL POSSIBLE ILLEGAL DSA
20001 ;          8974  !           ADDRESSES AND TEST THE
20002 ;          8975  !           IAE BIT SET.
20003 ;          8976  !
20004 ;          8977  ! IMPLICIT INPUTS:
20005 ;          8978  !
20006 ;          8979  !       IO_BUF
20007 ;          8980  !       A VECTOR OF 256 WORDS WHERE
20008 ;          8981  !       DATA FOR MBUS READS AND WRITE
20009 ;          8982  !       FUNCTION ARE FOUND.
20010 ;          8983  !
20011 ;          8984  !
20012 ;          8985  !--
20013 ;          8986
20014 ;          8987  local
20015 ;          8988      IAE_CNT;                                  !INVLID ADRS COUNT
20016 ;          8989
20017 ;          8990  IAE_CNT = .LST_BLK;                          !START AT LAST  BLOCK + 1
20018 ;          8991
20019 ;          8992  do                                          !TEST FOR ALL INVALID ADDRESSES
20020 ;          8993      begin
20021 ;          8994      IAE_CNT = .IAE_CNT + 1;                 !INCREMENT IAE_CNT
20022 ;          8995      BGNSUB;
20023 ;          8996      CLR_MBUS;
20024 ;          8997      MLDA = .IAE_CNT;                        !LOAD DSA
20025 ;          8998      MLWC = not 255;                         !LOAD WORD COUNT
20026 ;          8999      MLBA = IO_BUF;                          !LOAD UBUS ADRS
20027 ;          9000      MLCS1 = write;                          !DO A WRITE FUNCTION
20028 ;          9001
20029 ;          9002      if .IAE IS_NOT_SET                      !SET IF IAE SET
20030 ;          9003      then
20031 ;          9004          begin                               !ERROR IF NOT SET
20032 ;          9005          ERRDF (102, ASYNC, DUMPER);
20033 ;          9006          PRINTB (FIV_FMT, WRD_30, PHR_1, WRD_11, WRD_30, WRD_10);
20034 ;          9007          end;
20035 ;          9008
20036 ;          9009      ENDSUB;
```

```
20038 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
20039 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (94)
20040
20041 ;          9010      end
20042 ;          9011  until .IAE_CNT eql %o'177777';                !REPEAT UNTIL ALL TESTED
20043 ;          9012
20044 ;          9013  ENDTST;
20048
20049                                        .SBTTL  $T44
20053 071242  010146                 $T44:   MOV     R1,-(SP)                ;                              8956
20054 071244  016701  120322                 MOV     LST.BLK,R1              ; *,IAE.CNT                    8990
20055 071250  005201                 1$:     INC     R1                      ; IAE.CNT                      8994
20056 071252  104402                 2$:     TRAP    2
20057 071254  152777  000040  121732         BISB    #40,@ML.REG+40          ;                              8995
20058 071262  016700  122314                 MOV     ML.DUT,R0
20059 071266  042700  177770                 BIC     #177770,R0
20060 071272  142777  000007  121714         BICB    #7,@ML.REG+40
20061 071300  150077  121710                 BISB    R0,@ML.REG+40
20062 071304  010177  121674                 MOV     R1,@ML.REG+30           ; IAE.CNT,*                    8997
20063 071310  012777  177400  121646         MOV     #-400,@ML.REG+10        ;                              8998
20064 071316  012777  011600  121650         MOV     #IO.BUF,@ML.REG+20      ;                              8999
20065 071324  012777  000061  121622         MOV     #61,@ML.REG             ;                              9000
20066 071332  032777  002000  121674         BIT     #2000,@ML.REG+60        ;                              9002
20067 071340  001026                         BNE     3$
20068 071342  104455                         TRAP    55                      ;                              9005
20069 071344  000146                         .WORD   146
20070 071346  010464                         .WORD   ASYNC
20071 071350  024060                         .WORD   DUMPER
20072 071352  012746  006406                 MOV     #WRD.10,-(SP)           ;                              9006
20073 071356  012746  006624                 MOV     #WRD.30,-(SP)
20074 071362  012746  006416                 MOV     #WRD.11,-(SP)
20075 071366  012746  007366                 MOV     #PHR.1,-(SP)
20076 071372  012746  006624                 MOV     #WRD.30,-(SP)
20077 071376  012746  006062                 MOV     #FIV.FMT,-(SP)
20078 071402  012746  000006                 MOV     #6,-(SP)
20079 071406  010600                         MOV     SP,R0                   ; SP,*
20080 071410  104414                         TRAP    14
20081 071412  062706  000016                 ADD     #16,SP                  ;                              9004
20082 071416  104467                 3$:     TRAP    67                      ;                              9007
20083 071420  006000                         ROR     R0
20084 071422  103713                         BLO     2$
20085 071424  020127  177777                 CMP     R1,#-1                  ; IAE.CNT,*                    9011
20086 071430  001307                         BNE     1$
20087 071432  012601                         MOV     (SP)+,R1                ;                              8956
20088 071434  000207                         RTS     PC
20089
20090                                ; Routine Size:  62 words
20091                                ; Maximum stack depth per invocation:  8 words
```

```
20100
20101
20105
20106                                   .SBTTL  T44
20110 071436                    T44::
20111 071436  004767  177600  1$:     JSR     PC,$T44                 ;       9011
20112 071442  104466                  TRAP    66
20113 071444  006000                  ROR     R0
20114 071446  103773                  BLO     1$
20115 071450  000207                  RTS     PC
20116
20117                           ; Routine Size:  6 words
20118                           ; Maximum stack depth per invocation:  0 words
20123
20124
20125 ;         9014  !<BLF/PAGE>
```

```
20127 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
20128 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (95)
20129
20130 ;          9015  !
20131 ;          9016  BGNTST;
20132 ;          9017
20133 ;          9018  !++
20134 ;          9019  ! TEST NUMBER:  TST 45
20135 ;          9020  !
20136 ;          9021  ! TEST NAME:    ADRS OVERFLOW BIT TEST
20137 ;          9022  !
20138 ;          9023  ! TEST DESCRIPTION:
20139 ;          9024  !       TEST THE DETECTION OF ADDRESS OVERFLOWS BY:
20140 ;          9025  !
20141 ;          9026  !       1.  STARTING AT THE LAST BLOCK DO A TWO BLOCK TRANSFER.
20142 ;          9027  !
20143 ;          9028  !       2.  READ THE AOE BIT SET.
20144 ;          9029  !
20145 ;          9030  ! IMPLICIT INPUTS:
20146 ;          9031  !       IO_BUF
20147 ;          9032  !       A VECTOR OF 256 WORDS WHERE DATA FOR MBUS READS AND WRITE
20148 ;          9033  !       FUNCTIONS ARE FOUND.
20149 ;          9034  !--
20150 ;          9035
20151 ;          9036  CLR_MBUS;
20152 ;          9037  MLWC = not 511;                          !DO TWO BLOCK XFERR
20153 ;          9038  MLBA = IO_BUF;                           !LOAD UBUS ADRS
20154 ;          9039  MLDA = .LST_BLK;                         !START AT LAST BLOCK
20155 ;          9040  MLCS1 = write;                           !DO A WRITE FUNCTION
20156 ;          9041  TIME_OUT_LOOP;
20157 ;          9042
20158 ;          9043  if .AOE IS_NOT_SET                       !SEE IF AOE SET
20159 ;          9044  then
20160 ;          9045      begin                                !ERROR IF NOT SET
20161 ;          9046      ERRDF (103, SYNC, DUMPER);
20162 ;          9047      PRINTB (FOR_FMT, WRD_26, PHR_1, WRD_11, FNC_19);
20163 ;          9048      end;
20164 ;          9049
20165 ;          9050  ENDTST;
20169
20170                                          .SBTTL  $T45
20174 071452  152777  000040  121534  $T45:   BISB    #40,@ML.REG+40          ;                    9016
20175 071460  016700  122116                  MOV     ML.DUT,R0
20176 071464  042700  177770                  BIC     #177770,R0
20177 071470  142777  000007  121516          BICB    #7,@ML.REG+40
20178 071476  150077  121512                  BISB    R0,@ML.REG+40
20179 071502  012777  177000  121454          MOV     #-1000,@ML.REG+10       ;                    9037
20180 071510  012777  011600  121456          MOV     #IO.BUF,@ML.REG+20      ;                    9038
20181 071516  016777  120050  121460          MOV     LST.BLK,@ML.REG+30      ;                    9039
```

```
20183                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
20184                                   ;                                                       14-May-1981 13:05:08    PA:<
20185
20186 071524  012777  000061  121422            MOV     #61,@ML.REG                     ;                               9040
20187 071532  105777  121466           1$:      TSTB    @ML.REG+50
20188 071536  100375                            BPL     1$
20189 071540  032777  001000  121466            BIT     #1000,@ML.REG+60                ;                               9043
20190 071546  001024                            BNE     2$
20191 071550  104455                            TRAP    55                              ;                               9046
20192 071552  000147                            .WORD   147
20193 071554  010526                            .WORD   SYNC
20194 071556  024060                            .WORD   DUMPER
20195 071560  012746  010204                    MOV     #FNC.19,-(SP)                   ;                               9047
20196 071564  012746  006416                    MOV     #WRD.11,-(SP)
20197 071570  012746  007366                    MOV     #PHR.1,-(SP)
20198 071574  012746  006574                    MOV     #WRD.26,-(SP)
20199 071600  012746  006046                    MOV     #FOR.FMT,-(SP)
20200 071604  012746  000005                    MOV     #5,-(SP)
20201 071610  010600                            MOV     SP,R0                           ; SP,*
20202 071612  104414                            TRAP    14
20203 071614  062706  000014                    ADD     #14,SP                          ;                               9045
20204 071620  000207           2$:      RTS     PC                              ;                               9013
20205
20206                                   ; Routine Size:  52 words
20207                                   ; Maximum stack depth per invocation:  6 words
20212
20213
20217
20218                                           .SBTTL  T45
20222 071622                            T45::
20223 071622  004767  177624           1$:      JSR     PC,$T45                         ;                               9048
20224 071626  104466                            TRAP    66
20225 071630  006000                            ROR     R0
20226 071632  103773                            BLO     1$
20227 071634  000207                            RTS     PC
20228
20229                                   ; Routine Size:  6 words
20230                                   ; Maximum stack depth per invocation:  0 words
20235
20236
20237 ;          9051   !<BLF/PAGE>
```

```
20239 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
20240 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (96)
20241
20242 ;          9052  !
20243 ;          9053  BGNTST;
20244 ;          9054
20245 ;          9055  !++
20246 ;          9056  ! TEST NUMBER:  TST 46
20247 ;          9057  !
20248 ;          9058  ! TEST NAME:    SYNC BUS PARITY TEST
20249 ;          9059  !
20250 ;          9060  ! TEST DESCRIPTION:
20251 ;          9061  !       TEST ABILITY OF SYNC BUS TO DETECT AND GENERATE
20252 ;          9062  !       GOOD PARITY BY:
20253 ;          9063  !
20254 ;          9064  !       1.  VIA MBUS WRITE FUNCTION WRITE ALTERNATING ONES AND ZEROES TO THE
20255 ;          9065  !           DEVICE AND READ THE DPAR BIT CLEARED.
20256 ;          9066  !
20257 ;          9067  !       2.  REPEAT WITH SHIFTED DATA
20258 ;          9068  !
20259 ;          9069  !       3.  VIA MBUS READ FUNCTION READ THE ALTERNATING PATTERN AND MDPE CLEARED.
20260 ;          9070  !
20261 ;          9071  ! IMPLICIT INPUTS:
20262 ;          9072  !       IO_BUF
20263 ;          9073  !       A VECTOR OF 256 WORDS WHERE DATA FOR MBUS READS AND WRITES
20264 ;          9074  !       FUNCTIONS ARE FOUND.
20265 ;          9075  !--
20266 ;          9076
20267 ;          9077  if .PAR_DIS IS_SET                          !SEE IF PARITY IS DISABLED
20268 ;          9078  then
20269 ;          9079      begin                                   !PRINT MESSAGE AND EXIT TST IF YES
20270 ;          9080      PRINTB (THR_FMT, FNC_3, WRD_7, WRD_37);
20271 ;          9081      EXIT_TST;
20272 ;          9082      end;
20273 ;          9083
20274 ;          9084  CLR_MBUS;
20275 ;          9085  BAI = ONE;                                  !SET ON FIRST IO_BUF ADRS
20276 ;          9086  IO_BUF = %o'125252';                        !ALTERNATE 1, 0 PATTERN
20277 ;          9087
20278 ;          9088  incr TWICE from 0 to 1 do                   !REPEAT LOOP TWICE
20279 ;          9089      begin
20280 ;          9090      BGNSUB;
20281 ;          9091      GD_BLK_XFER ();                         !SET UP A GOOD BLOCK XFERR
20282 ;          9092      MLCS1 = write;                          !DO A WRITE FUNCTION
20283 ;          9093      TIME_OUT_LOOP;
20284 ;          9094
20285 ;          9095      if .DPAR IS_SET                         !SEE IF DPAR GOT SET
20286 ;          9096      then
20287 ;          9097          begin                               !ERROR IF SET
20288 ;          9098          ERRDF (105, SYNC, DUMPER);
20289 ;          9099          PRINTB (FOR_FMT, WRD_23, WRD_6, WRD_7, WRD_9);
20290 ;          9100          end;
20291 ;          9101
20292 ;          9102      IO_BUF = .IO_BUF^ONE;                   !SHIFT THE IO_BUF & REPEAT
20293 ;          9103      MLER = ZEROES;                          !CLEAR OUT ERROR REG & REPEAT
```

```
20295 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
20296 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (96)
20297
20298 ;          9104      ENDSUB;
20299 ;          9105      end;
20300 ;          9106
20301 ;          9107  incr TWICE from 0 to 1 do                    !REPEATE LOOP TWICE
20302 ;          9108      begin
20303 ;          9109      BGNSUB;
20304 ;          9110      CLR_MBUS;
20305 ;          9111      BAI = ONE;                                !SET ON FIRST IO_BUF ADRS
20306 ;          9112      GD_BLK_XFER ();                           !SET UP A GOOD BLOCK XFERR
20307 ;          9113      MLCS1 = read;                             !DO A READ XFERR
20308 ;          9114      TIME_OUT_LOOP;
20309 ;          9115
20310 ;          9116      if .MDPE IS_SET                           !SEE IF READ GENERATED BAD PARITY
20311 ;          9117      then
20312 ;          9118          begin                                 !ERROR IF MDPE SET
20313 ;          9119          ERRDF (106, SYNC, DUMPER);
20314 ;          9120          PRINTB (FOR_FMT, WRD_23, WRD_6, WRD_7, WRD_8);
20315 ;          9121          end;
20316 ;          9122
20317 ;          9123      ENDSUB;
20318 ;          9124      IO_BUF = .IO_BUF^ONE;                     !MAKE DATA PATTERN HAVE ONE LESS ONE
20319 ;          9125      GD_BLK_XFER ();                           !SET UP A GOOD BLK XFER
20320 ;          9126      MLCS1 = write;                            !CHANGE THE BACKGROUND IN MEMORY
20321 ;          9127      TIME_OUT_LOOP;
20322 ;          9128      end;
20323 ;          9129
20324 ;          9130  ENDTST;
20328
20329                                        .SBTTL  $T46
20333 071636  010146                 $T46:  MOV     R1,-(SP)          ;                          9050
20334 071640  026727  117722  000001        CMP     PAR.DIS,#1        ;                          9077
20335 071646  001021                        BNE     1$
20336 071650  012746  006700                MOV     #WRD.37,-(SP)     ;                          9080
20337 071654  012746  006344                MOV     #WRD.7,-(SP)
20338 071660  012746  007750                MOV     #FNC.3,-(SP)
20339 071664  012746  006034                MOV     #THR.FMT,-(SP)
20340 071670  012746  000004                MOV     #4,-(SP)
20341 071674  010600                        MOV     SP,R0             ; SP,*
20342 071676  104414                        TRAP    14
20343 071700  104463                        TRAP    63
20344 071702  062706  000012                ADD     #12,SP            ;                          9077
20345 071706  000167  000402                JMP     9$                ;                          9079
20346 071712  152777  000040  121274  1$:   BISB    #40,@ML.REG+40    ;                          9082
20347 071720  016701  121656                MOV     ML.DUT,R1
20348 071724  042701  177770                BIC     #177770,R1
20349 071730  142777  000007  121256        BICB    #7,@ML.REG+40
```

```
                                                                                                14-May-1981 13:48:54    TOPS
                                                                                                14-May-1981 13:05:08    PA:<
20351                           ;ML4
20352                           ;
20353
20354 071736  150177  121252               BISB    R1,@ML.REG+40
20355 071742  152777  000010  121244       BISB    #10,@ML.REG+40          ;                                       9085
20356 071750  012767  125252  117622       MOV     #-52526,IO.BUF          ;                                       9086
20357 071756  005001                       CLR     R1                      ; TWICE                                 9088
20358 071760  104402                 2$:   TRAP    2                       ;                                       9089
20359 071762  004767  124102               JSR     PC,GD.BLK.XFER          ;                                       9091
20360 071766  012777  000061  121160       MOV     #61,@ML.REG             ;                                       9092
20361 071774  105777  121224           3$: TSTB    @ML.REG+50
20362 072000  100375                       BPL     3$
20363 072002  132777  000040  121224       BITB    #40,@ML.REG+60          ;                                       9095
20364 072010  001424                       BEQ     4$
20365 072012  104455                       TRAP    55                      ;                                       9098
20366 072014  000151                       .WORD   151
20367 072016  010526                       .WORD   SYNC
20368 072020  024060                       .WORD   DUMPER
20369 072022  012746  006374               MOV     #WRD.9,-(SP)            ;                                       9099
20370 072026  012746  006344               MOV     #WRD.7,-(SP)
20371 072032  012746  006336               MOV     #WRD.6,-(SP)
20372 072036  012746  006544               MOV     #WRD.23,-(SP)
20373 072042  012746  006046               MOV     #FOR.FMT,-(SP)
20374 072046  012746  000005               MOV     #5,-(SP)
20375 072052  010600                       MOV     SP,R0                   ; SP,*
20376 072054  104414                       TRAP    14
20377 072056  062706  000014               ADD     #14,SP                  ;                                       9097
20378 072062  006367  117512           4$: ASL     IO.BUF                  ;                                       9102
20379 072066  005077  121142               CLR     @ML.REG+60              ;                                       9103
20380 072072  104467                       TRAP    67
20381 072074  006000                       ROR     R0
20382 072076  103730                       BLO     2$
20383 072100  005201                       INC     R1                      ; TWICE                                 9088
20384 072102  020127  000001               CMP     R1,#1                   ; TWICE,*
20385 072106  003724                       BLE     2$
20386 072110  005001                       CLR     R1                      ; TWICE                                 9107
20387 072112  104402                 5$:   TRAP    2                       ;                                       9108
20388 072114  152777  000040  121072       BISB    #40,@ML.REG+40          ;                                       9109
20389 072122  016700  121454               MOV     ML.DUT,R0
20390 072126  042700  177770               BIC     #177770,R0
20391 072132  142777  000007  121054       BICB    #7,@ML.REG+40
20392 072140  150077  121050               BISB    R0,@ML.REG+40
20393 072144  152777  000010  121042       BISB    #10,@ML.REG+40          ;                                       9111
20394 072152  004767  123712               JSR     PC,GD.BLK.XFER          ;                                       9112
20395 072156  012777  000071  120770       MOV     #71,@ML.REG             ;                                       9113
20396 072164  105777  121034           6$: TSTB    @ML.REG+50
20397 072170  100375                       BPL     6$
20398 072172  032777  000400  121014       BIT     #400,@ML.REG+40         ;                                       9116
20399 072200  001424                       BEQ     7$
20400 072202  104455                       TRAP    55                      ;                                       9119
20401 072204  000152                       .WORD   152
20402 072206  010526                       .WORD   SYNC
20403 072210  024060                       .WORD   DUMPER
20404 072212  012746  006360               MOV     #WRD.8,-(SP)            ;                                       9120
20405 072216  012746  006344               MOV     #WRD.7,-(SP)
```

```
20407                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
20408                                   ;                                                       14-May-1981 13:05:08    PA:<
20409
20410 072222  012746  006336                     MOV     #WRD.6,-(SP)
20411 072226  012746  006544                     MOV     #WRD.23,-(SP)
20412 072232  012746  006046                     MOV     #FOR.FMT,-(SP)
20413 072236  012746  000005                     MOV     #5,-(SP)
20414 072242  010600                             MOV     SP,R0                          ; SP,*
20415 072244  104414                             TRAP    14
20416 072246  062706  000014                     ADD     #14,SP                         ;                                       9118
20417 072252  104467                  7$:        TRAP    67                             ;                                       9121
20418 072254  006000                             ROR     R0
20419 072256  103715                             BLO     5$
20420 072260  006367  117314                     ASL     IO.BUF                         ;                                       9124
20421 072264  004767  123600                     JSR     PC,GD.BLK.XFER                 ;                                       9125
20422 072270  012777  000061  120656             MOV     #61,@ML.REG                    ;                                       9126
20423 072276  105777  120722            8$:      TSTB    @ML.REG+50
20424 072302  100375                             BPL     8$
20425 072304  005201                             INC     R1                             ; TWICE                                 9107
20426 072306  020127  000001                     CMP     R1,#1                          ; TWICE,*
20427 072312  003677                             BLE     5$
20428 072314  012601                  9$:        MOV     (SP)+,R1                       ;                                       9050
20429 072316  000207                             RTS     PC
20430
20431                                   ; Routine Size:  153 words
20432                                   ; Maximum stack depth per invocation:  7 words
20437
20438
20442
20443                                           .SBTTL  T46
20447 072320                          T46::
20448 072320  004767  177312            1$:     JSR     PC,$T46                        ;                                       9128
20449 072324  104466                             TRAP    66
20450 072326  006000                             ROR     R0
20451 072330  103773                             BLO     1$
20452 072332  000207                             RTS     PC
20453
20454                                   ; Routine Size:  6 words
20455                                   ; Maximum stack depth per invocation:  0 words
20460
20461 ;          9131  !<BLF/PAGE>
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (97)

```
20463 ;ML4
20464 ;
20465
20466 ;          9132  !
20467 ;          9133  !
20468 ;          9134  BGNTST;
20469 ;          9135
20470 ;          9136  !++
20471 ;          9137  ! TEST NUMBER:  TST 47
20472 ;          9138  !
20473 ;          9139  ! TEST NAME:    WRITE READ ML11
20474 ;          9140  !
20475 ;          9141  ! TEST DESCRIPTION:
20476 ;          9142  !
20477 ;          9143  !       PROVIDES A MBUS READ/WRITE
20478 ;          9144  !       DATA TRANSFER TROUBLE SHOOTING
20479 ;          9145  !       LOOP BY:
20480 ;          9146  !
20481 ;          9147  !       1.  LOAD APPROPRIATE RH REGISTERS.
20482 ;          9148  !           DO A WRITE FUNCTION.
20483 ;          9149  !
20484 ;          9150  !       2.  LOAD APPROPRIATE RH REGISTERS
20485 ;          9151  !           DO A WRITE CHECK FUNCTION.
20486 ;          9152  !
20487 ;          9153  !       3.  COMPLIMENT DATA AND
20488 ;          9154  !           REPEAT.
20489 ;          9155  !
20490 ;          9156  ! IMPLICIT INPUTS:
20491 ;          9157  !
20492 ;          9158  !       IO_BUF
20493 ;          9159  !       A VECTOR OF 256 WORDS WHERE
20494 ;          9160  !       DATA FOR MBUS READ AND WRITE
20495 ;          9161  !       FUNCTIONS IS FOUND.
20496 ;          9162  !
20497 ;          9163  !--
20498 ;          9164
20499 ;          9165  local
20500 ;          9166      TST_PAT;                          !TEST PATTERN
20501 ;          9167
20502 ;          9168  TST_PAT = %o'052525';
20503 ;          9169
20504 ;          9170  incr TWICE from 0 to 1 do             !WRITE READ 1'S AND 0'S ON MBUS
20505 ;          9171      begin
20506 ;          9172      CLR_MBUS;
20507 ;          9173      BAI = ONE;                        !SET ON FIRST IO_BUF ADRS
20508 ;          9174      IO_BUF = .TST_PAT;                !LOAD FIRST IO_BUF ADRS
20509 ;          9175      GD_BLK_XFER ();                   !SET UP A GOOD BLOCK XFERR
20510 ;          9176      MLCS1 = write;                    !DO A WRITE FUNCTION
20511 ;          9177      TIME_OUT_LOOP;
20512 ;          9178      CLR_MBUS;
20513 ;          9179      BAI = ONE;                        !SET ON FIRST IO_BUF ADRS
20514 ;          9180      GD_BLK_XFER ();                   !SET UP A GOOD BLOCK XFERR
20515 ;          9181      MLCS1 = WRT_CHK;                  !DO A WRITE CHECK FUNCTION
20516 ;          9182      TIME_OUT_LOOP;
20517 ;          9183
```

```
20519 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
20520 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (97)
20521
20522 ;          9184      if .WCE IS_SET                     !SEE IF WRITE CHECK ERROR SET
20523 ;          9185      then
20524 ;          9186          begin                          !ERROR IF SET
20525 ;          9187          ERRDF (109, TRBLE_LOOP, DUMPER);
20526 ;          9188          PRINTB (SIX_FMT, FNC_4, WRD_10, WRD_12, FNC_5, FNC_6, FNC_3);
20527 ;          9189          end;
20528 ;          9190
20529 ;          9191      TST_PAT = not .TST_PAT;            !COMPLIMENT TST_PAT AND REPEAT
20530 ;          9192      end;
20531 ;          9193
20532 ;          9194  ENDTST;
20536
20537                                          .SBTTL  $T47
20541 072334  004167  111504          $T47:   JSR     R1,$SAVE2                ;                          9130
20542 072340  012701  052525                  MOV     #52525,R1                ; *.TST.PAT                9168
20543 072344  005002                          CLR     R2                       ; TWICE                    9170
20544 072346  152777  000040  120640  1$:     BISB    #40,@ML.REG+40           ;                          9171
20545 072354  016700  121222                  MOV     ML.DUT,R0
20546 072360  042700  177770                  BIC     #177770,R0
20547 072364  142777  000007  120622          BICB    #7,@ML.REG+40
20548 072372  150077  120616                  BISB    R0,@ML.REG+40
20549 072376  152777  000010  120610          BISB    #10,@ML.REG+40           ;                          9173
20550 072404  010167  117170                  MOV     R1,IO.BUF                ; TST.PAT,*                9174
20551 072410  004767  123454                  JSR     PC,GD.BLK.XFER           ;                          9175
20552 072414  012777  000061  120532          MOV     #61,@ML.REG              ;                          9176
20553 072422  105777  120576          2$:     TSTB    @ML.REG+50
20554 072426  100375                          BPL     2$
20555 072430  152777  000040  120556          BISB    #40,@ML.REG+40           ;                          9177
20556 072436  016700  121140                  MOV     ML.DUT,R0
20557 072442  042700  177770                  BIC     #177770,R0
20558 072446  142777  000007  120540          BICB    #7,@ML.REG+40
20559 072454  150077  120534                  BISB    R0,@ML.REG+40
20560 072460  152777  000010  120526          BISB    #10,@ML.REG+40           ;                          9179
20561 072466  004767  123376                  JSR     PC,GD.BLK.XFER           ;                          9180
20562 072472  012777  000051  120454          MOV     #51,@ML.REG              ;                          9181
20563 072500  105777  120520          3$:     TSTB    @ML.REG+50
20564 072504  100375                          BPL     3$
20565 072506  032777  040000  120500          BIT     #40000,@ML.REG+40        ;                          9184
20566 072514  001430                          BEQ     4$
20567 072516  104455                          TRAP    55                       ;                          9187
20568 072520  000155                          .WORD   155
20569 072522  011026                          .WORD   TRBLE.LOOP
20570 072524  024060                          .WORD   DUMPER
20571 072526  012746  007750                  MOV     #FNC.3,-(SP)             ;                          9188
20572 072532  012746  010004                  MOV     #FNC.6,-(SP)
20573 072536  012746  007774                  MOV     #FNC.5,-(SP)
```

```
20575                           ;ML4                                                    14-May-1981 13:48:54    TOPS
20576                           ;                                                       14-May-1981 13:05:08    PA:<
20577
20578 072542  012746  006426              MOV     #WRD.12,-(SP)
20579 072546  012746  006406              MOV     #WRD.10,-(SP)
20580 072552  012746  007756              MOV     #FNC.4,-(SP)
20581 072556  012746  006100              MOV     #SIX.FMT,-(SP)
20582 072562  012746  000007              MOV     #7,-(SP)
20583 072566  010600                      MOV     SP,R0                           ; SP,*
20584 072570  104414                      TRAP    14
20585 072572  062706  000020              ADD     #20,SP                          ;                       9186
20586 072576  005101              4$:     COM     R1                              ; TST.PAT               9191
20587 072600  005202                      INC     R2                              ; TWICE                 9170
20588 072602  020227  000001              CMP     R2,#1                           ; TWICE,*
20589 072606  003657                      BLE     1$
20590 072610  000207                      RTS     PC                              ;                       9130
20591
20592                           ; Routine Size:  87 words
20593                           ; Maximum stack depth per invocation:  11 words
20598
20599
20603
20604                                   .SBTTL  T47
20608 072612                    T47::
20609 072612  004767  177516    1$:     JSR     PC,$T47                         ;                       9192
20610 072616  104466                      TRAP    66
20611 072620  006000                      ROR     R0
20612 072622  103773                      BLO     1$
20613 072624  000207                      RTS     PC
20614
20615                           ; Routine Size:  6 words
20616                           ; Maximum stack depth per invocation:  0 words
20621
20622
20623 ;          9195  !<BLF/PAGE>
```

```
20625 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
20626 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (98)
20627
20628 ;          9196  !
20629 ;          9197  BGNTST;
20630 ;          9198
20631 ;          9199  !++
20632 ;          9200  ! TEST NUMBER:  TST 48
20633 ;          9201  !
20634 ;          9202  ! TEST NAME:    CRC DATA BUS TEST
20635 ;          9203  !
20636 ;          9204  ! TEST DESCRIPTION:
20637 ;          9205  !       TEST THE CRC DATA BUS BETWEEN THE CRC
20638 ;          9206  !       GENERATORS AND THE CRC/MBUS DATA
20639 ;          9207  !       MUX FOR CONTINUITY AND BIT UNIQUENESS
20640 ;          9208  !       BY:
20641 ;          9209  !
20642 ;          9210  !       1.  WHILE IN ECC DIAGNOSTIC MODE DO
20643 ;          9211  !           GOOD BLOCK TRANSFERS WITH VARIOUS
20644 ;          9212  !           CRC_A,CRC_B AND PAR_CRC_WRD WHICH
20645 ;          9213  !           WILL EXERCISE BUS CONTINUITY AND
20646 ;          9214  !           UNIQUENESS.
20647 ;          9215  !
20648 ;          9216  !       2.  THEN IN DATA DIAGNOSTEC MODE READ
20649 ;          9217  !           THE CRC NIBBLES FROM ONE CRC GROUP
20650 ;          9218  !           FOR CORRECT CRC_A,CRC_B AND PAR_CRC_WRD
20651 ;          9219  !           DATA PATTERNS.
20652 ;          9220  !
20653 ;          9221  ! IMPLICIT INPUTS:
20654 ;          9222  !       PD_TEMP
20655 ;          9223  !       A BIT VECTOR OF 16 BITS WHERE THE READ
20656 ;          9224  !       PROM DATA IS STORED OAND ACCESSED FROM.
20657 ;          9225  !
20658 ;          9226  !
20659 ;          9227  !--
20660 ;          9228
20661 ;          9229  local
20662 ;          9230      CRC_NIB,                          !POINTER TO THE CRC NIBBLE
20663 ;          9231      CRC_TSTED,                        !STORES HOW MANY CRC NIBBLES WE HAVE TESTED
20664 ;          9232      ERR_FLG,                          !ERROR FLAG
20665 ;          9233      NIB_PAT;                          !TEST DATA FOR THE CRC NIBBLE
20666 ;          9234
20667 ;          9235  CRC_NIB = 9;                          !SET CRC NIBBLE POINTER
20668 ;          9236
20669 ;          9237  incr LOOP from 0 to 4 do              !REPEAT THE TEST WITH FIVE NIBBLE PATTERNS
20670 ;          9238      begin
20671 ;          9239      BGNSUB;                           !START OF THE SCOPE LOOP
20672 ;          9240      CLR_MBUS;
20673 ;          9241      ECC_DM = ONE;                     !SET ECC DIAG MOD
20674 ;          9242
20675 ;          9243      case .LOOP from 0 to 4 of         !SELECT ONE OF THE FIVE NIBBLE PATTERNS
20676 ;          9244          set
20677 ;          9245
20678 ;          9246          [0] :                         !PATTERN 0
20679 ;          9247              NIB_PAT = %b'110';
```

```
20681 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
20682 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (98)
20683
20684 ;          9248
20685 ;          9249          [1] :                                    !PATTERN 1
20686 ;          9250              begin
20687 ;          9251              MLE1 = ONES;
20688 ;          9252              MLE2 = ONES;
20689 ;          9253              NIB_PAT = %b'001';
20690 ;          9254              end;
20691 ;          9255
20692 ;          9256          [2] :                                    !PATTERN 2
20693 ;          9257              begin
20694 ;          9258              CRC_A = ONES;
20695 ;          9259              CRC_B = ONES;
20696 ;          9260              NIB_PAT = %b'000';
20697 ;          9261              end;
20698 ;          9262
20699 ;          9263          [3] :                                    !PATTERN 3
20700 ;          9264              begin
20701 ;          9265              PAR_CRC_WRD = ONES;
20702 ;          9266              CRC_B = ONES;
20703 ;          9267              NIB_PAT = %b'011';
20704 ;          9268              end;
20705 ;          9269
20706 ;          9270          [4] :                                    !PATTERN 4
20707 ;          9271              begin
20708 ;          9272              PAR_CRC_WRD = ONES;
20709 ;          9273              CRC_A = ONES;
20710 ;          9274              NIB_PAT = %b'101';
20711 ;          9275              end;
20712 ;          9276          tes;
20713 ;          9277
20714 ;          9278          GD_BLK_XFER ();                          !SET UP A GOOD BLOCK TRANSFER
20715 ;          9279          MLCS1 = write;                           !DO A WRITE TRANSFER
20716 ;          9280          TIME_OUT_LOOP;                           !WAIT UNTIL DRIVE READY
20717 ;          9281          DAT_DM = ONE;                            !SET DATA DIAG MODE
20718 ;          9282          GD_BLK_XFER ();                          !SET UP A GOOD BLOCK TRANSFER
20719 ;          9283          MLCS1 = read;                            !DO A READ TRANSFER
20720 ;          9284          CRC_TSTED = ZERO;
20721 ;          9285          DELAY (ONE_US);
20722 ;          9286
20723 ;          9287          do                                       !TEST ONE CRC GROUP '6 CRC NIBBLES'
20724 ;          9288              begin
20725 ;          9289              PD_TEMP = .MLPD;                     !READ THE PROM DATA FOR THIS ARRAY WORD
20726 ;          9290              DAT_CLK = ONE;                       !CLOCK OUT AN ARRAY WORD INTO THE DATA DIAG REG'S
20727 ;          9291              DELAY (ONE_US);                      !DELAY ONE MICRO SECOND
20728 ;          9292              RD_LNG_WRD;                          !READ THE DATA DIAG REG'S INTO MEMORY
20729 ;          9293
20730 ;          9294              if .PD_TEMP [.CRC_NIB] IS_NOT_SET    !TEST THIS CRC NIBBLE IF GOOD
20731 ;          9295              then
20732 ;          9296                  begin
20733 ;          9297                  CRC_TSTED = .CRC_TSTED + 1;      !INCREMENT THE GOOD NIBBLE COUNTER
20734 ;          9298                  TST_LNG_WRD (.CRC_NIB, .NIB_PAT, ERR_FLG);  !TEST THE CRC NIBBLE
20735 ;          9299
```

```
20737 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
20738 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (98)
20739
20740 ;          9300              if .ERR_FLG IS_SET                    !SEE IF THE TEST FOUND AN ERROR
20741 ;          9301              then
20742 ;          9302                  begin                             !REPORT ERROR IF FLAG IS SET
20743 ;          9303                  ERRDF (131, SYNC, DUMPER);
20744 ;          9304                  PRINTB (FOR_FMT, WRD_64, WRD_24, WRD_73, WRD_10);
20745 ;          9305                  PRINTB (FMT_5, .NIB_PAT, .NIB_SAVE [NIB_9], 8);
20746 ;          9306                  end;
20747 ;          9307
20748 ;          9308              end;
20749 ;          9309
20750 ;          9310          end
20751 ;          9311      until .CRC_TSTED eql 6;                       !REPEAT UNTIL ONE CRC GROUP HAS BEEN TESTED
20752 ;          9312
20753 ;          9313      ENDSUB;                                       !END OF SCOPE LOOP
20754 ;          9314      end;
20755 ;          9315
20756 ;          9316  ENDTST;
20760
20761                                      .SBTTL  $T48
20765 072626  004167  111264          $T48:  JSR     R1,$SAVE5                ;                          9194
20766 072632  162706  000006                 SUB     #6,SP
20767 072636  012705  000011                 MOV     #11,R5                   ; *,CRC.NIB                9235
20768 072642  010546                         MOV     R5,-(SP)                 ; CRC.NIB,*                9294
20769 072644  006216                         ASR     (SP)
20770 072646  006216                         ASR     (SP)
20771 072650  006216                         ASR     (SP)
20772 072652  062716  013120                 ADD     #PD.TEMP,(SP)
20773 072656  005001                         CLR     R1                       ; LOOP                     9237
20774 072660  104402                   1$:   TRAP    2                        ;                          9238
20775 072662  152777  000040  120324        BISB    #40,@ML.REG+40           ;                          9239
20776 072670  016704  120706                 MOV     ML.DUT,R4
20777 072674  042704  177770                 BIC     #177770,R4
20778 072700  142777  000007  120306        BICB    #7,@ML.REG+40
20779 072706  150477  120302                 BISB    R4,@ML.REG+40
20780 072712  152777  000001  120354        BISB    #1,@ML.REG+120           ;                          9241
20781 072720  010104                         MOV     R1,R4                    ; LOOP,*                   9243
20782 072722  006304                         ASL     R4
20783 072724  066407  072730                 ADD     2$(R4),PC
20784 072730  000012                   2$:   .WORD   3$-2$
20785 072732  000020                         .WORD   4$-2$
20786 072734  000042                         .WORD   5$-2$
20787 072736  000062                         .WORD   6$-2$
20788 072740  000104                         .WORD   7$-2$
20789 072742  012702  000006           3$:   MOV     #6,R2                    ; *,NIB.PAT                9247
20790 072746  000437                         BR      8$                       ;                          9243
20791 072750  012777  177777  120346  4$:   MOV     #-1,@ML.REG+150          ;                          9251
```

```
                                                                                                                    H 1
                                ;ML4                                                                    14-May-1981 13:48:54    TOPS
                                :                                                                       14-May-1981 13:05:08    PA:<
                                MOV     #-1,@ML.REG+160         ;                                                               9252
                                MOV     #1,R2                   ; *,NIB.PAT                                                     9253
                                BR      8$                      ;                                                               9243
                        5$:     BISB    #77,@ML.REG+150         ;                                                               9258
                                BISB    #77,@ML.REG+160         ;                                                               9259
                                CLR     R2                      ; NIB.PAT                                                       9260
                                BR      8$                      ;                                                               9243
                        6$:     BIS     #37400,@ML.REG+150      ;                                                               9265
                                BISB    #77,@ML.REG+160         ;                                                               9266
                                MOV     #3,R2                   ; *,NIB.PAT                                                     9267
                                BR      8$                      ;                                                               9243
                        7$:     BIS     #177777,@ML.REG+150     ;                                                               9272
                                MOV     #5,R2                   ; *,NIB.PAT                                                     9274
                        8$:     JSR     PC,GD.BLK.XFER          ;                                                               9278
                                MOV     #61,@ML.REG             ;                                                               9279
                        9$:     TSTB    @ML.REG+50
                                BPL     9$
                                BISB    #10,@ML.REG+120         ;                                                               9281
                                JSR     PC,GD.BLK.XFER          ;                                                               9282
                                MOV     #71,@ML.REG             ;                                                               9283
                                CLR     2(SP)                   ; CRC.TSTED                                                     9284
                                MOV     #1,R3                   ; *,$$TMP2                                                      9285
                        10$:    BEQ     13$
                                MOV     L$DLY,R4                ; *,$$TMP1
                                BEQ     12$
                        11$:    CLR     6(SP)                   ; $$TMP
                                DEC     R4                      ; $$TMP1
                                BNE     11$
                        12$:    DEC     R3                      ; $$TMP2
                                BR      10$
                        13$:    MOV     @ML.REG+230,PD.TEMP     ;                                                               9289
                                BISB    #20,@ML.REG+120         ;                                                               9290
                                MOV     #1,R3                   ; *,$$TMP2                                                      9291
                        14$:    BEQ     17$
                                MOV     L$DLY,R4                ; *,$$TMP1
                                BEQ     16$
                        15$:    CLR     6(SP)                   ; $$TMP
                                DEC     R4                      ; $$TMP1
                                BNE     15$
                        16$:    DEC     R3                      ; $$TMP2
                                BR      14$
                        17$:    MOV     @ML.REG+170,D1.TEMP
                                MOV     @ML.REG+200,D2.TEMP
                                MOV     @ML.REG+160,E2.TEMP
                                MOV     (SP),-(SP)              ;                                                               9294
                                MOV     R5,-(SP)                ; CRC.NIB,*
                                BIC     #177770,(SP)
                                MOV     #1,-(SP)
                                CLR     -(SP)
                                JSR     PC,BL$GT2
                                ADD     #10,SP
```

```
20848                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
20849                                   ;                                                       14-May-1981 13:05:08    PA:<
20850
20851 073256  005700                            TST     R0
20852 073260  001066                            BNE     19$
20853 073262  005266  000002                    INC     2(SP)                   ; CRC.TSTED                             9297
20854 073266  010546                            MOV     R5,-(SP)                ; CRC.NIB,*                             9298
20855 073270  010246                            MOV     R2,-(SP)                ; NIB.PAT,*
20856 073272  012746  000012                    MOV     #12,-(SP)
20857 073276  060616                            ADD     SP,(SP)                 ; ERR.FLG,*
20858 073300  004767  122650                    JSR     PC,TST.LNG.WRD
20859 073304  026627  000012  000001            CMP     12(SP),#1               ; ERR.FLG,*                             9300
20860 073312  001047                            BNE     18$
20861 073314  104455                            TRAP    55                      ;                                       9303
20862 073316  000203                            .WORD   203
20863 073320  010526                            .WORD   SYNC
20864 073322  024060                            .WORD   DUMPER
20865 073324  012746  006406                    MOV     #WRD.10,-(SP)           ;                                       9304
20866 073330  012746  007310                    MOV     #WRD.73,-(SP)
20867 073334  012746  006552                    MOV     #WRD.24,-(SP)
20868 073340  012746  007224                    MOV     #WRD.64,-(SP)
20869 073344  012746  006046                    MOV     #FOR.FMT,-(SP)
20870 073350  012746  000005                    MOV     #5,-(SP)
20871 073354  010600                            MOV     SP,R0                   ; SP,*
20872 073356  1044**                            TRAP    14
20873 073360  012716  000010                    MOV     #10,(SP)                ;                                       9305
20874 073364  016704  115566                    MOV     NIB.SAVE+4,R4
20875 073370  006204                            ASR     R4
20876 073372  006204                            ASR     R4
20877 073374  006204                            ASR     R4
20878 073376  006204                            ASR     R4
20879 073400  000304                            SWAB    R4
20880 073402  042704  177770                    BIC     #177770,R4
20881 073406  010446                            MOV     R4,-(SP)
20882 073410  010246                            MOV     R2,-(SP)                ; NIB.PAT,*
20883 073412  012746  004430                    MOV     #FMT.5,-(SP)
20884 073416  012746  000004                    MOV     #4,-(SP)
20885 073422  010600                            MOV     SP,R0                   ; SP,*
20886 073424  104414                            TRAP    14
20887 073426  062706  000024                    ADD     #24,SP                  ;                                       9302
20888 073432  062706  000006          18$:      ADD     #6,SP                   ;                                       9296
20889 073436  026627  000002  000006  19$:      CMP     2(SP),#6                ; CRC.TSTED,*                           9311
20890 073444  001236                            BNE     13$
20891 073446  104467                            TRAP    67
20892 073450  006000                            ROR     R0
20893 073452  103002                            BHIS    21$
20894 073454  000167  177200          20$:      JMP     1$
20895 073460  005201                21$:      INC     R1                      ; LOOP                                  9237
20896 073462  020127  000004                    CMP     R1,#4                   ; LOOP,*
20897 073466  003772                            BLE     20$
20898 073470  062706  000010                    ADD     #10,SP                  ;                                       9194
20899 073474  000207                            RTS     PC
20900
20901                                   ; Routine Size:  215 words
20902                                   ; Maximum stack depth per invocation:  23 words
```

```
20904                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
20905                                   ;                                                       14-May-1981 13:05:08    PA:<
20906
20911
20912
20916
20917                                           .SBTTL  T48
20921 073476                            T48::
20922 073476  004767  177124            1$:     JSR     PC,$T48                 ;                               9314
20923 073502  104466                            TRAP    66
20924 073504  006000                            ROR     R0
20925 073506  103773                            BLO     1$
20926 073510  000207                            RTS     PC
20927
20928                                   ; Routine Size:  6 words
20929                                   ; Maximum stack depth per invocation:  0 words
20934
20935
20936 ;          9317  !<BLF/PAGE>
```

```
20938 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
20939 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (99)
20940
20941 ;          9318  !
20942 ;          9319  BGNTST
20943 ;          9320
20944 ;          9321  !++
20945 ;          9322  ! TEST NUMBER:  TST 49
20946 ;          9323  !
20947 ;          9324  ! TEST NAME:    UNIQUE NIBBLE CRC GENERATION TEST
20948 ;          9325  !
20949 ;          9326  ! TEST DESCRIPTION:
20950 ;          9327  !       TEST CRC CODES GENERATED FOR ONE CRC
20951 ;          9328  !       GROUP (52 UNIQUE UNIBUS NIBBLES) BY:
20952 ;          9329  !
20953 ;          9330  !       1.  DOING MASS BUS WRITE TRANSFERSS WITH THE
20954 ;          9331  !       FIRST IO_BUF CRC GROUP HAVING 51 NIBBLES
20955 ;          9332  !       OF %B'0000' PATTERN AND THE REMAINING NIBBLE
20956 ;          9333  !           WITH PATTERNS OF %B'0000' TO %B'1111'.
20957 ;          9334  !
20958 ;          9335  !       2.  THEN IN DATA DIAGNOSTIC MODE CALCULATE
20959 ;          9336  !           THE CRC CODE FOR THE FIRST CRC
20960 ;          9337  !           GROUP AND COMPARE IT AGAINST THE
20961 ;          9338  !           HARDWARE GENERATED CRC CODE.
20962 ;          9339  !
20963 ;          9340  !       3.  REPEAT PATTERN OF %B'0000' TO %B'1111'
20964 ;          9341  !           FOR EACH OF THE 52 NIBBLES.
20965 ;          9342  !
20966 ;          9343  ! IMPLICIT INPUTS:
20967 ;          9344  !       IO_BUF
20968 ;          9345  !       A  VECTOR OF 256 WORDS WHERE DATA
20969 ;          9346  !       FOR MBUS READS AND WRITES TRANSFERS
20970 ;          9347  !       IS FOUND.
20971 ;          9348  !
20972 ;          9349  !
20973 ;          9350  !--
20974 ;          9351
20975 ;          9352  local
20976 ;          9353  DODU_FLG ,                                  !DROP UNIT FLAG
20977 ;          9354  TEMP ,                                      !TEMPORARY STORAGE LOCATION
20978 ;          9355  NIB_SEL;                                    !SELECTS 4 BITS 'ONE NIBBLE' FROM THE IO_BUF
20979 ;          9356  DODU_FLG = ZEROES;                          !CLEAR THE DROP UNIT FLAG
20980 ;          9357
20981 ;          9358  incr CNT from 0 to 13 do                    !CLEAR THE FIRST 14 IO_BUF WORDS
20982 ;          9359      IO_BUF [.CNT] = ZEROES;
20983 ;          9360
20984 ;          9361  incr WRD_CNT from 0 to 12 do                !DO THIS TEST ON 13 IO_BUF WORDS
20985 ;          9362      begin
20986 ;          9363      NIB_SEL = -4;                           !RESET THE NIBBLE SELECTOR
20987 ;          9364
20988 ;          9365      incr NIB_TST from 0 to 3 do             !DO THIS TEST ON 4 NIBBLES PER WORD
20989 ;          9366          begin
20990 ;          9367
20991 ;          9368          if (.WRD_CNT eql 12) and (.NIB_TST eql 2) then exitloop;        !THE LAST 2 NIBBLES ARE NOT TESTED
20992 ;          9369
```

```
20994 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
20995 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (99)
20996
20997 ;          9370        NIB_SEL = .NIB_SEL + 4;                  !INCREMENT THE NIBBLE SELECTED
20998 ;          9371        TEMP = ZEROES;                           !CLEAR  TEMP LOCATION
20999 ;          9372
21000 ;          9373        incr NIB_PAT from %b'0000' to %b'1111' by %b'0001' do   !RUN PATTERNS ON THEIS NIBBLE
21001 ;          9374            begin
21002 ;          9375            BGNSUB;                              !START OF SCOPE LOOP
21003 ;          9376            TEMP<.NIB_SEL, 4> = .NIB_PAT;        !LOAD PATTERN INTO TEMP AT THIS NIBBLE
21004 ;          9377            IO_BUF [.WRD_CNT] = .TEMP;           !LOAD THE GENERATED PATTERN INTO THE IO_BUF
21005 ;          9378            CLR_MBUS;
21006 ;          9379            GD_BLK_XFER ();                      !SET UP A GOOD BLOCK TRANSFER
21007 ;          9380            MLCS1 = write;                       !DO A WRITE TRANSFER
21008 ;          9381            TIME_OUT_LOOP;                       !WAIT FOR THE TRANSFER TO COMPLETE
21009 ;          9382            A_GEN = ZEROES;                      !CLEAR THE GEN & CAL STORAGE LOCATIONS
21010 ;          9383            B_GEN = ZEROES;
21011 ;          9384            P_GEN = ZEROES;
21012 ;          9385            A_CAL = ZEROES;
21013 ;          9386            B_CAL = ZEROES;
21014 ;          9387            P_CAL = ZEROES;
21015 ;          9388            CAL_CRC ();                          !CALL ROUTINE TO CALCULATE THE CRC CODE FOR THIS PATTERN
21016 ;          9389
21017 ;          9390            if ERR_CHK_CRC ()                    !NOW SEE IF THE GEN CRC IS SAME AS CAL CRC
21018 ;          9391            then
21019 ;          9392                begin                            !REPORT THE ERROR IF NOT THE SAME
21020 ;          9393                ERRDF (132, SYNC, DUMPER);
21021 ;          9394                PRINTB (THR_FMT, WRD_5, WRD_64, WRD_8);
21022 ;          9395                PRINTB (FMT_19, .B_GEN, .A_GEN, .P_GEN);
21023 ;          9396                PRINTB (FMT_20, .B_CAL, .A_CAL, .P_CAL);
21024 ;          9397                DODU_FLG = ONE;
21025 ;          9398                end;
21026 ;          9399
21027 ;          9400            ENDSUB;                              !END OF SCOPE LOOP
21028 ;          9401            end;
21029 ;          9402
21030 ;          9403          IO_BUF [.WRD_CNT] = ZEROES;            !CLEAR THIS !O_BUF WORD FO NEXT NIBBLE
21031 ;          9404          end;
21032 ;          9405
21033 ;          9406      end;
21034 ;          9407
21035 ;          9408  if .DODU_FLG IS_SET                           !DROP THIS !NIT IF THE DODU FLAG GOT SET
21036 ;          9409  then
21037 ;          9410      begin
21038 ;          9411      DODU (.ML_LUN);
21039 ;          9412      DOCLN;
21040 ;          9413      end;
21041 ;          9414
21042 ;          9415  ENDTST;
21046
21047                                  .SBTTL  $T49
```

```
21055 073512  004167  110400          $T49:   JSR     R1,$SAVE5                       ;                                       9316
21056 073516  024646                          CMP     -(SP),-(SP)
21057 073520  005066  000002                  CLR     2(SP)                           ; DODU.FLG                              9356
21058 073524  005000                          CLR     R0                              ; CNT                                   9358
21059 073526  010001                  1$:     MOV     R0,R1                           ; CNT,*                                 9359
21060 073530  006301                          ASL     R1
21061 073532  005061  011600                  CLR     IO.BUF(R1)
21062 073536  005200                          INC     R0                              ; CNT                                   9358
21063 073540  020027  000015                  CMP     R0,#15                          ; CNT,*
21064 073544  003770                          BLE     1$
21065 073546  005002                          CLR     R2                              ; WRD.CNT                               9361
21066 073550  012716  177774          2$:     MOV     #-4,(SP)                        ; *,NIB.SEL                             9363
21067 073554  010201                          MOV     R2,R1                           ; WRD.CNT,*                             9377
21068 073556  006301                          ASL     R1
21069 073560  005004                          CLR     R4                              ; NIB.TST                               9365
21070 073562  020227  000014          3$:     CMP     R2,#14                          ; WRD.CNT,*                             9368
21071 073566  001003                          BNE     4$
21072 073570  020427  000002                  CMP     R4,#2                           ; NIB.TST,*
21073 073574  001562                          BEQ     8$
21074 073576  062716  000004          4$:     ADD     #4,(SP)                         ; *,NIB.SEL                             9370
21075 073602  005005                          CLR     R5                              ; TEMP                                  9371
21076 073604  005003                          CLR     R3                              ; NIB.PAT                               9373
21077 073606  104402                  5$:     TRAP    2                               ;                                       9374
21078 073610  010546                          MOV     R5,-(SP)                        ; TEMP,*                                9376
21079 073612  016646  000002                  MOV     2(SP),-(SP)                     ; NIB.SEL,*
21080 073616  012746  000004                  MOV     #4,-(SP)
21081 073622  010346                          MOV     R3,-(SP)                        ; NIB.PAT,*
21082 073624  004767  107452                  JSR     PC,BL$PU1
21083 073630  010005                          MOV     R0,R5                           ; *,TEMP                                9374
21084 073632  010561  011600                  MOV     R5,IO.BUF(R1)                   ; TEMP,*                                9377
21085 073636  152777  000040  117350          BISB    #40,@ML.REG+40
21086 073644  016700  117732                  MOV     ML.DUT,R0
21087 073650  042700  177770                  BIC     #177770,R0
21088 073654  142777  000007  117332          BICB    #7,@ML.REG+40
21089 073662  150077  117326                  BISB    R0,@ML.REG+40
21090 073666  004767  122176                  JSR     PC,GD.BLK.XFER                  ;                                       9379
21091 073672  012777  000061  117254          MOV     #61,@ML.REG                     ;                                       9380
21092 073700  105777  117320          6$:     TSTB    @ML.REG+50
21093 073704  100375                          BPL     6$
21094 073706  005067  117234                  CLR     A.GEN                           ;                                       9382
21095 073712  005067  117232                  CLR     B.GEN                           ;                                       9383
21096 073716  005067  117230                  CLR     P.GEN                           ;                                       9384
21097 073722  005067  117212                  CLR     A.CAL                           ;                                       9385
21098 073726  005067  117210                  CLR     B.CAL                           ;                                       9386
21099 073732  005067  117206                  CLR     P.CAL                           ;                                       9387
21100 073736  004767  120650                  JSR     PC,CAL.CRC                      ;                                       9388
21101 073742  004767  121244                  JSR     PC,ERR.CHK.CRC                  ;                                       9390
21102 073746  006000                          ROR     R0
21103 073750  103055                          BCC     7$
21104 073752  104455                          TRAP    55                              ;                                       9393
```

```
21106                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
21107                                   ;                                                       14-May-1981 13:05:08    PA:<
21108
21109 073754  000204                            .WORD   204
21110 073756  010526                            .WORD   SYNC
21111 073760  024060                            .WORD   DUMPER
21112 073762  012746  006360                    MOV     #WRD.8,-(SP)            ;                                       9394
21113 073766  012746  007224                    MOV     #WRD.64,-(SP)
21114 073772  012746  006330                    MOV     #WRD.5,-(SP)
21115 073776  012746  006034                    MOV     #THR.FMT,-(SP)
21116 074002  012746  000004                    MOV     #4,-(SP)
21117 074006  010600                            MOV     SP,R0                   ; SP,*
21118 074010  104414                            TRAP    14
21119 074012  016716  117134                    MOV     P.GEN,(SP)              ;                                       9395
21120 074016  016746  117124                    MOV     A.GEN,-(SP)
21121 074022  016746  117122                    MOV     B.GEN,-(SP)
21122 074026  012746  005412                    MOV     #FMT.19,-(SP)
21123 074032  012746  000004                    MOV     #4,-(SP)
21124 074036  010600                            MOV     SP,R0                   ; SP,*
21125 074040  104414                            TRAP    14
21126 074042  016716  117076                    MOV     P.CAL,(SP)              ;                                       9396
21127 074046  016746  117066                    MOV     A.CAL,-(SP)
21128 074052  016746  117064                    MOV     B.CAL,-(SP)
21129 074056  012746  005462                    MOV     #FMT.20,-(SP)
21130 074062  012746  000004                    MOV     #4,-(SP)
21131 074066  010600                            MOV     SP,R0                   ; SP,*
21132 074070  104414                            TRAP    14
21133 074072  012766  000001  000044            MOV     #1,44(SP)               ; *,DODU.FLG                            9397
21134 074100  062706  000032                    ADD     #32,SP                  ;                                       9392
21135 074104  062706  000010          7$:       ADD     #10,SP                  ;                                       9374
21136 074110  104467                            TRAP    67                      ;                                       9398
21137 074112  006000                            ROR     R0
21138 074114  103634                            BLO     5$
21139 074116  005203                            INC     R3                      ; NIB.PAT                               9373
21140 074120  020327  000017                    CMP     R3,#17                  ; NIB.PAT,*
21141 074124  003630                            BLE     5$
21142 074126  005061  011600                    CLR     IO.BUF(R1)              ;                                       9403
21143 074132  005204                            INC     R4                      ; NIB.TST                               9365
21144 074134  020427  000003                    CMP     R4,#3                   ; NIB.TST,*
21145 074140  003610                            BLE     3$
21146 074142  005202                    8$:     INC     R2                      ; WRD.CNT                               9361
21147 074144  020227  000014                    CMP     R2,#14                  ; WRD.CNT,*
21148 074150  003002                            BGT     9$
21149 074152  000167  177372                    JMP     2$
21150 074156  026627  000002  000001    9$:     CMP     2(SP),#1                ; DODU.FLG,*                            9408
21151 074164  001004                            BNE     10$
21152 074166  016700  117406                    MOV     ML.LUN,R0               ;                                       9411
21153 074172  104451                            TRAP    51
21154 074174  104444                            TRAP    44
21155 074176  022626                    10$:    CMP     (SP)+,(SP)+             ;                                       9316
21156 074200  000207                            RTS     PC
21157
21158                                   ; Routine Size:  156 words
21159                                   ; Maximum stack depth per invocation:  25 words
```

```
21168
21169
21173
21174                                           .SBTTL  T49
21178 074202                            T49::
21179 074202  004767  177304            1$:     JSR     PC,$T49                 ;                       9413
21180 074206  104466                            TRAP    66
21181 074210  006000                            ROR     R0
21182 074212  103773                            BLO     1$
21183 074214  000207                            RTS     PC
21184
21185                                   ; Routine Size:  6 words
21186                                   ; Maximum stack depth per invocation:  0 words
21191
21192
21193 ;         9416  !
21194 ;         9417  !<BLF/PAGE>
```

```
21196 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
21197 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (100)
21198
21199 ;          9418  !
21200 ;          9419  BGNTST;
21201 ;          9420
21202 ;          9421  !++
21203 ;          9422  ! TEST NUMBER:  TST 50
21204 ;          9423  !
21205 ;          9424  ! TEST NAME:            UNIQUE WORD CRC GENERATION TEST
21206 ;          9425  !
21207 ;          9426  ! TEST DESCRIPTION:
21208 ;          9427  !       TEST CRC CODES GENERATED FOR ONE
21209 ;          9428  !       CRC GROUP (13 UNIBUS WORDS) BY:
21210 ;          9429  !
21211 ;          9430  !       1.  DOING MASS BUS WRITE TRANSFERS
21212 ;          9431  !           USING VARIOUS WORD DATA PATTERENS.
21213 ;          9432  !
21214 ;          9433  !       2.  THEN IN DATA DIAGNOSTIC MODE
21215 ;          9434  !           CALCULATE THE CRC CODE FOR THE
21216 ;          9435  !           FIRST CRC GROUP AND COMPARE IT TO
21217 ;          9436  !           THE HARDWARE GENERATED CRC CODE.
21218 ;          9437  !
21219 ;          9438  ! IMPLICIT INPUTS:
21220 ;          9439  !       IO_BUF
21221 ;          9440  !       A  VECTOR OF 256 WORDS WHERE DATA
21222 ;          9441  !       FOR MBUS READS AND WRITES TRANSFERS
21223 ;          9442  !       IS FOUND.
21224 ;          9443  !
21225 ;          9444  !--
21226 ;          9445
21227 ;          9446  local
21228 ;          9447      DODU_FLG,                          !DROP UNIT FLAG
21229 ;          9448      TST_PAT;                           !TEST PATTERN SAVE LOCATION
21230 ;          9449
21231 ;          9450  DODU_FLG = ZEROES;                     !CLEAR THE DROP UNIT FLAG
21232 ;          9451
21233 ;          9452  incr PAT_SEL from 0 to 3 do            !RUN THE TEST ON 4 PATTERNS
21234 ;          9453      begin
21235 ;          9454      BGNSUB;                            !START OF SCOPE LOOP
21236 ;          9455      CLR_MBUS;
21237 ;          9456
21238 ;          9457      case .PAT_SEL from 0 to 3 of       !SELECT ONE OF THE 4 PATTERNS
21239 ;          9458          set
21240 ;          9459
21241 ;          9460          [0] :
21242 ;          9461              TST_PAT = ONES;
21243 ;          9462
21244 ;          9463          [1] :
21245 ;          9464              TST_PAT = %o'052525';
21246 ;          9465
21247 ;          9466          [2] :
21248 ;          9467              TST_PAT = %o'125252';
21249 ;          9468
21250 ;          9469          [3] :
21251 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
21252 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (100)
```

```
21253
21254 ;          9470              TST_PAT = %o'070707';
21255 ;          9471          tes;
21256 ;          9472
21257 ;          9473      incr CNT from 0 to 255 do                    !LOAD THE FIRST 13 IO_BUF WORDS WITH TST_PAT
21258 ;          9474          IO_BUF [.CNT] = .TST_PAT;
21259 ;          9475
21260 ;          9476      GD_BLK_XFER ();                              !SET UP A GOOD BLOCK TRANSFER
21261 ;          9477      ML[CS1 = write;                              !DO A WRITE TRANSFER
21262 ;          9478      TIME_OUT_LOOP;                               !WAIT FOR THE TRANSFER TO COMPLETE
21263 ;          9479      A_GEN = ZEROES;                              !CLEAR THE GEN & CAL SAVE LOCATIONS
21264 ;          9480      B_GEN = ZEROES;
21265 ;          9481      P_GEN = ZEROES;
21266 ;          9482      A_CAL = ZEROES;
21267 ;          9483      B_CAL = ZEROES;
21268 ;          9484      P_CAL = ZEROES;
21269 ;          9485      CAL_CRC ();                                  !CALCULATE WHAT THE CRC CODE SHOULD BE
21270 ;          9486
21271 ;          9487      if ERR_CHK_CRC ()                            !NOW SEE IF THE GENERATED CRC CODE IS THE SAME
21272 ;          9488      then
21273 ;          9489          begin                                    !REPORT THE ERROR IF NOT THE SAME
21274 ;          9490          ERRDF (133, SYNC, DUMPER);
21275 ;          9491          PRINTB (THR_FMT, WRD_5, WRD_64, WRD_8);
21276 ;          9492          PRINTB (FMT_19, .B_GEN, .A_GEN, .P_GEN);
21277 ;          9493          PRINTB (FMT_20, .B_CAL, .A_CAL, .P_CAL);
21278 ;          9494          DODU_FLG = ONE;
21279 ;          9495          end;
21280 ;          9496
21281 ;          9497      ENDSUB;                                      !END OF SCOPE LOOP
21282 ;          9498      end;
21283 ;          9499
21284 ;          9500  if .DODU_FLG IS_SET                              !DROP THIS UNIT IF THE DODU FLAG GOT SET
21285 ;          9501  then
21286 ;          9502      begin
21287 ;          9503      DODU (.ML_LUN);
21288 ;          9504      DOCLN;
21289 ;          9505      end;
21290 ;          9506
21291 ;          9507  ENDTST;
21295
21296                                        .SBTTL  $T50
21300 074216  004167  107674          $T50: JSR     R1,$SAVE5                  ;                                  9415
21301 074222  005005                      CLR     R5                         ; DODU.FLG                         9450
21302 074224  005004                      CLR     R4                         ; PAT.SEL                          9452
21303 074226  104402                1$:   TRAP    2                          ;                                  9453
21304 074230  152777  000040  116756      BISB    #40,@ML.REG+40             ;                                  9454
21305 074236  016703  117340              MOV     ML.DUT,R3
21306                               ;ML4                                                  14-May-1981 13:48:54  TOPS
21307                               ;                                                     14-May-1981 13:05:08  PA:<
21308
21309 074242  042703  177770              BIC     #177770,R3
21310 074246  142777  000007  116740      BICB    #7,@ML.REG+40
21311 074254  150377  116734              BISB    R3,@ML.REG+40
21312 074260  010403                      MOV     R4,R3                      ; PAT.SEL,*                        9457
21313 074262  006303                      ASL     R3
21314 074264  066307  074270              ADD     2$(R3),PC
21315 074270  000010                2$:   .WORD   3$-2$
```

```
21316 074272  000016                    .WORD   4$-2$
21317 074274  000024                    .WORD   5$-2$
21318 074276  000032                    .WORD   6$-2$
21319 074300  012701  177777      3$:   MOV     #-1,R1          ; *,TST.PAT                     9461
21320 074304  000410                    BR      7$              ;                               9457
21321 074306  012701  052525      4$:   MOV     #52525,R1       ; *,TST.PAT                     9464
21322 074312  000405                    BR      7$              ;                               9457
21323 074314  012701  125252      5$.   MOV     #-52526,R1      ; *,TST.PAT                     9467
21324 074320  000402                    BR      7$              ;                               9457
21325 074322  012701  070707      6$:   MOV     #70707,R1       ; *,TST.PAT                     9470
21326 074326  005002              7$:   CLR     R2              ; CNT                           9473
21327 074330  010203              8$:   MOV     R2,R3           ; CNT,*                         9474
21328 074332  006303                    ASL     R3
21329 074334  010163  011600            MOV     R1,IO.BUF(R3)   ; TST.PAT,*
21330 074340  005202                    INC     R2              ; CNT                           9473
21331 074342  020227  000377            CMP     R2,#377         ; CNT,*
21332 074346  003770                    BLE     8$
21333 074350  004767  121514            JSR     PC,GD.BLK.XFER  ;                               9476
21334 074354  012777  000061  116572    MOV     #61,@ML.REG     ;                               9477
21335 074362  105777  116636      9$:   TSTB    @ML.REG+50
21336 074366  100375                    BPL     9$
21337 074370  005067  116552            CLR     A.GEN           ;                               9479
21338 074374  005067  116550            CLR     B.GEN           ;                               9480
21339 074400  005067  116546            CLR     P.GEN           ;                               9481
21340 074404  005067  116530            CLR     A.CAL           ;                               9482
21341 074410  005067  116526            CLR     B.CAL           ;                               9483
21342 074414  005067  116524            CLR     P.CAL           ;                               9484
21343 074420  004767  120166            JSR     PC,CAL.CRC      ;                               9485
21344 074424  004767  120562            JSR     PC,ERR.CHK.CRC  ;                               9487
21345 074430  006000                    ROR     R0
21346 074432  103054                    BCC     10$
21347 074434  104455                    TRAP    55              ;                               9490
21348 074436  000205                    .WORD   205
21349 074440  010526                    .WORD   SYNC
21350 074442  024060                    .WORD   DUMPER
21351 074444  012746  006360            MOV     #WRD.8,-(SP)    ;                               9491
21352 074450  012746  007224            MOV     #WRD.64,-(SP)
21353 074454  012746  006330            MOV     #WRD.5,-(SP)
21354 074460  012746  006034            MOV     #THR.FMT,-(SP)
21355 074464  012746  000004            MOV     #4,-(SP)
21356 074470  010600                    MOV     SP,R0           ; SP,*
21357 074472  104414                    TRAP    14
21358 074474  016716  116452            MOV     P.GEN,(SP)      ;                               9492
21359 074500  016746  116442            MOV     A.GEN,-(SP)
21360 074504  016746  116440            MOV     B.GEN,-(SP)
21361                       ;ML4                                  14-May-1981 13:48:54    TOPS
21362                       ;                                     14-May-1981 13:05:08    PA:<
21363
21364 074510  012746  005412            MOV     #FMT.19,-(SP)
21365 074514  012746  000004            MOV     #4,-(SP)
21366 074520  010600                    MOV     SP,R0           ; SP,*
21367 074522  104414                    TRAP    14
21368 074524  016716  116414            MOV     P.CAL,(SP)      ;                               9493
21369 074530  016746  116404            MOV     A.CAL,-(SP)
21370 074534  016746  116402            MOV     B.CAL,-(SP)
21371 074540  012746  005462            MOV     #FMT.20,-(SP)
21372 074544  012746  000004            MOV     #4,-(SP)
```

```
21373 074550  010600                    MOV     SP,R0                           ; SP,*
21374 074552  104414                    TRAP    14
21375 074554  012705  000001            MOV     #1,R5                           ; *,DODU.FLG                    9494
21376 074560  062706  000032            ADD     #32,SP                          ;                               9489
21377 074564  104467            10$:    TRAP    67                              ;                               9495
21378 074566  006000                    ROR     R0
21379 074570  103616                    BLO     1$
21380 074572  005204                    INC     R4                              ; PAT.SEL                       9452
21381 074574  020427  000003            CMP     R4,#3                           ; PAT.SEL,*
21382 074600  003612                    BLE     1$
21383 074602  005305                    DEC     R5                              ; DODU.FLG                      9500
21384 074604  001004                    BNE     11$
21385 074606  016700  116766            MOV     ML.LUN,R0                       ;                               9503
21386 074612  104451                    TRAP    51
21387 074614  104444                    TRAP    44
21388 074616  000207            11$:    RTS     PC                              ;                               9415
21389
21390                           ; Routine Size:  129 words
21391                           ; Maximum stack depth per invocation:  19 words
21396
21397
21401
21402                                   .SBTTL  T50
21406 074620                    T50::
21407 074620  004767  177372    1$:     JSR     PC,$T50                         ;                               9505
21408 074624  104466                    TRAP    66
21409 074626  006000                    ROR     R0
21410 074630  103773                    BLO     1$
21411 074632  000207                    RTS     PC
21412
21413                           ; Routine Size:  6 words
21414                           ; Maximum stack depth per invocation:  0 words
21422
21423
21424 ;          9508  !<BLF/PAGE>
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (101)

```
21426 ;ML4
21427 ;
21428
21429 ;          9509  !
21430 ;          9510  BGNTST;
21431 ;          9511
21432 ;          9512  !++
21433 ;          9513  ! TEST NUMBER:  TST 51
21434 ;          9514  !
21435 ;          9515  ! TEST NAME:            CORRECTABLE ERROR SYNDRONE DECODE TEST
21436 ;          9516  !
21437 ;          9517  ! TEST DESCRIPTION:
21438 ;          9518  !       TEST SYNDRONE DECODE AND ERROR
21439 ;          9519  !       CORRECTION TO DECODE AND CORRECT
21440 ;          9520  !       SINGLE BIT AND MULTIPLE BIT
21441 ;          9521  !       CHANNEL ERRORS BY:
21442 ;          9522  !
21443 ;          9523  !       1.  VIA FCC DIAG MODE FORCE SYNDRONE
21444 ;          9524  !           BITS TO DECODE SINGLE BIT AND
21445 ;          9525  !           MULTIPLE BIT CHANNEL ERRORS.
21446 ;          9526  !
21447 ;          9527  !       2.  THEN DO A MASS BUS READ TRANSFER
21448 ;          9528  !           AND EXAM THE IO_BUF FOR CORRECT
21449 ;          9529  !           BIT COMPLIMENTING.
21450 ;          9530  !
21451 ;          9531  ! IMPLICIT INPUTS:
21452 ;          9532  !       IO_BUF
21453 ;          9533  !       A  VECTOR OF 256 WORDS WHERE DATA
21454 ;          9534  !       FOR MBUS READS AND WRITES TRANSFERS
21455 ;          9535  !       IS FOUND.
21456 ;          9536  !
21457 ;          9537  !
21458 ;          9538  !--
21459 ;          9539
21460 ;          9540  local
21461 ;          9541      BITS_XFERED;                    !POINTS TO THE CHANNEL BEING TESTED
21462 ;          9542
21463 ;          9543  BAI = ONE;                          !BEFORE WE START LETS CLEAR OUT THE ML11'S
21464 ;          9544  IO_BUF = ZEROES;                    !GOOD BLOCK BY WRITTING ZEROES TO IT
21465 ;          9545  GD_BLK_XFER ();
21466 ;          9546  ML[S1 = write;
21467 ;          9547  TIME_OUT_LOOP;
21468 ;          9548  BITS_XFERED = -1;                   !RESET THE CHANNEL POINTER
21469 ;          9549
21470 ;          9550  incr PLOG from 0 to 5 do            !TEST ONE CRC GROUP '6 ARRAY WORDS'
21471 ;          9551      begin
21472 ;          9552
21473 ;          9553      incr CHANNEL from 0 to 35 do    !TEST 36 BITS IN EACH ARRAY WORD
21474 ;          9554          begin
21475 ;          9555          BGNSUB;                     !START OF SCOPE LOOP
21476 ;          9556          CLR_MBUS;                   !CLEAR THE DRIVE
21477 ;          9557          BITS_XFERED = .BITS_XFERED + 1;   !INCREMENT THE CHANNEL POINTER
21478 ;          9558
21479 ;          9559          incr CNT from 0 to 255 do   !CLEAR THE FIRST CRC GROUP IN THE IO_BUF
21480 ;          9560              IO_BUF [.CNT] = ZEROES;
```

```
21482 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
21483 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (101)
21484
21485 ;          9561
21486 ;          9562          FORCE_REM (.PLOG, .CHANNEL);              !FORCE THIS BIT IN THIS WORD IN ERROR
21487 ;          9563          GD_BLK_XFER ();                           !SET UP A GOOD BLOCK TRANSFER
21488 ;          9564          ECC_DIS = ZERO;                           !CLEAR ECC DISABLE
21489 ;          9565          MLCS1 = read;                             !DO A READ TRANSFER
21490 ;          9566          TIME_OUT_LOOP;                            !WAIT FOR THE TRANSFER TO COMPLETE
21491 ;          9567
21492 ;          9568          if not FIND_COMP_BIT (.BITS_XFERED)       !SEARCH THE IO_BUF FOR THE COMP BIT
21493 ;          9569          then
21494 ;          9570              begin                                 !REPORT AN ERROR IF THIS BIT IS NOT COMP
21495 ;          9571              ERRDF (134, SYNC, DUMPER);
21496 ;          9572              PRINTB (THR_FMT, WRD_74, WRD_75, PHR_1);
21497 ;          9573              PRINTB (FMT_21, .PLOG, .CHANNEL);
21498 ;          9574              end;
21499 ;          9575
21500 ;          9576          incr WRD_CNT from 0 to 12 do              !SEE IF ANY OTHER BITS GOT COMP'ED
21501 ;          9577
21502 ;          9578              if .IO_BUF [.WRD_CNT] neq ZEROES      !TEST THE IO_BUF FOR ZEROES
21503 ;          9579              then
21504 ;          9580                  begin                             !REPORT AN ERROR IF ANY OTHERS ARE SET
21505 ;          9581                  ERRDF (135, SYNC, DUMPER);
21506 ;          9582                  PRINTB (FOR_FMT, WRD_61, WRD_74, WRD_75, PHR_5);
21507 ;          9583                  PRINTB (FMT_21, .PLOG, .CHANNEL);
21508 ;          9584                  end;
21509 ;          9585
21510 ;          9586              ENDSUB;                               !END OF SCOPE LOOP
21511 ;          9587              end;
21512 ;          9588
21513 ;          9589          end;
21514 ;          9590
21515 ;          9591      BITS_XFERED = -1;                             !RESET THE CHANNEL POINTER
21516 ;          9592
21517 ;          9593      incr CHANNEL from 0 to 35 do                  !TEST 36 CHANNELS FOR MULTIPLE BIT ERRORS
21518 ;          9594          begin
21519 ;          9595          BGNSUB;                                   !START OF SCOPE LOOP
21520 ;          9596          CLR_MBUS;                                 !CLEAR THE DRIVE
21521 ;          9597          BITS_XFERED = .BITS_XFERED + 1;           !INCREMENT THE CHANNEL POINTER
21522 ;          9598
21523 ;          9599          incr CNT from 0 to 20 do                  !CLEAR THE FIRST CRC GROUP IN THE IO_BUF
21524 ;          9600              IO_BUF [.CNT] = ZEROES;
21525 ;          9601
21526 ;          9602          FORCE_REM (58, .CHANNEL);                 !FORCE ALL 6 BITS IN THIS CHANNEL IN ERROR
21527 ;          9603          GD_BLK_XFER ();                           !SET UP A GOOD BLOCK TRANSFER
21528 ;          9604          ECC_DIS = ZERO;                           !CLEAR ECC DISABLE
21529 ;          9605          MLCS1 = read;                             !DO A READ TRANSFER
21530 ;          9606          TIME_OUT_LOOP;                            !WAIT FOR THE TRANSFER TO CMOPLETE
21531 ;          9607
21532 ;          9608          incr BIT_OFFSET from 0 to 180 by 36 do    !SEE IF EVERY 36TH BIT GOT COMP'ED
21533 ;          9609
21534 ;          9610              if not FIND_COMP_BIT (.BITS_XFERED + .BIT_OFFSET)       !SEARCH THE IO_BUF FOR THE COMP'ED BITS
21535 ;          9611              then
21536 ;          9612                  begin                             !REPORT AN ERROR IF ANY OF THE BITS ARE NOT COMP'ED
```

```
21538 :ML4                                                          14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
21539 :                                                             14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (101)
21540
21541 :          9613              ERRDF (134, SYNC, DUMPER);
21542 :          9614              PRINTB (THR_FMT, WRD_74, WRD_75, PHR_1);
21543 :          9615              PRINTB (FMT_21, 58, .CHANNEL);
21544 :          9616              end;
21545 :          9617
21546 :          9618      incr WRD_CNT from 0 to 12 do                !ALSO SEE IF ANY OTHER BIT GOT COMP'ED
21547 :          9619
21548 :          9620          if .IO_BUF [.WRD_CNT] neq ZEROES        !TEST THE IO_BUF FOR ALL ZEROES
21549 :          9621          then
21550 :          9622              begin                               !REPORT AN ERROR IF ANY OTHERS ARE COMP'ED
21551 :          9623              ERRDF (135, SYNC, DUMPER);
21552 :          9624              PRINTB (FIV_FMT, WRD_61, WRD_74, WRD_75, WRD_12, WRD_67);
21553 :          9625              PRINTB (FMT_21, 58, .CHANNEL);
21554 :          9626              end;
21555 :          9627
21556 :          9628      ENDSUB;                                     !END OF SCOPE LOOP
21557 :          9629      end;
21558 :          9630
21559 :          9631  ENDTST;
21563
21564                                          .SBTTL  $T51
21568 074634  004167  107256          $T51:   JSR     R1,$SAVE5                       ;                       9507
21569 074640  152777  000010  116346          BISB    #10,@ML.REG+40                  ;                       9543
21570 074646  005067  114726                  CLR     IO.BUF                          ;                       9544
21571 074652  004767  121212                  JSR     PC,GD.BLK.XFER                  ;                       9545
21572 074656  012777  000061  116270          MOV     #61,@ML.REG                     ;                       9546
21573 074664  105777  116334          1$:     TSTB    @ML.REG+50
21574 074670  100375                          BPL     1$
21575 074672  012705  177777                  MOV     #-1,R5                          ; *,BITS.XFERED         9548
21576 074676  005004                          CLR     R4                              ; PLOG                  9550
21577 074700  005003                  2$:     CLR     R3                              ; CHANNEL               9553
21578 074702  104402                  3$:     TRAP    2                               ;                       9554
21579 074704  152777  000040  116302          BISB    #40,@ML.REG+40                  ;                       9555
21580 074712  016702  116664                  MOV     ML.DUT,R2
21581 074716  042702  177770                  BIC     #177770,R2
21582 074722  142777  000007  116264          BICB    #7,@ML.REG+40
21583 074730  150277  116260                  BISB    R2,@ML.REG+40
21584 074734  005205                          INC     R5                              ; BITS.XFERED           9557
21585 074736  005001                          CLR     R1                              ; CNT                   9559
21586 074740  010102                  4$:     MOV     R1,R2                           ; CNT,*                 9560
21587 074742  006302                          ASL     R2
21588 074744  005062  011600                  CLR     IO.BUF(R2)
21589 074750  005201                          INC     R1                              ; CNT                   9559
21590 074752  020127  000377                  CMP     R1,#377                         ; CNT,*
21591 074756  003770                          BLE     4$
21592 074760  010446                          MOV     R4,-(SP)                        ; PLOG,*                9562
```

```
21594                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
21595                                   ;                                                       14-May-1981 13:05:08    PA:<
21596
21597 074762  010346                            MOV     R3,-(SP)                ; CHANNEL,*
21598 074764  004767  120612                    JSR     PC,FORCE.REM
21599 074770  004767  121074                    JSR     PC,GD.BLK.XFER          ;                                       9563
21600 074774  142777  000002  116272            BICB    #2,@ML.REG+120          ;                                       9564
21601 075002  012777  000071  116144            MOV     #71,@ML.REG             ;                                       9565
21602 075010  105777  116210          5$:       TSTB    @ML.REG+50
21603 075014  100375                            BPL     5$
21604 075016  010546                            MOV     R5,-(SP)                ; BITS.XFERED,*                         9568
21605 075020  004767  120404                    JSR     PC,FIND.COMP.BIT
21606 075024  005726                            TST     (SP)+
21607 075026  006000                            ROR     R0
21608 075030  103432                            BLO     6$
21609 075032  104455                            TRAP    55                      ;                                       9571
21610 075034  000206                            .WORD   206
21611 075036  010526                            .WORD   SYNC
21612 075040  024060                            .WORD   DUMPER
21613 075042  012746  007366                    MOV     #PHR.1,-(SP)            ;                                       9572
21614 075046  012746  007330                    MOV     #WRD.75,-(SP)
21615 075052  012746  007316                    MOV     #WRD.74,-(SP)
21616 075056  012746  006034                    MOV     #THR.FMT,-(SP)
21617 075062  012746  000004                    MOV     #4,-(SP)
21618 075066  010600                            MOV     SP,R0                   ; SP,*
21619 075070  104414                            TRAP    14
21620 075072  010316                            MOV     R3,(SP)                 ; CHANNEL,*                             9573
21621 075074  010446                            MOV     R4,-(SP)                ; PLOG,*
21622 075076  012746  005534                    MOV     #FMT.21,-(SP)
21623 075102  012746  000003                    MOV     #3,-(SP)
21624 075106  010600                            MOV     SP,R0                   ; SP,*
21625 075110  104414                            TRAP    14
21626 075112  062706  000020                    ADD     #20,SP                  ;                                       9570
21627 075116  005001                    6$:     CLR     R1                      ; WRD.CNT                               9576
21628 075120  010102                    7$:     MOV     R1,R2                   ; WRD.CNT,*                             9578
21629 075122  006302                            ASL     R2
21630 075124  005762  011600                    TST     IO.BUF(R2)
21631 075130  001434                            BEQ     8$
21632 075132  104455                            TRAP    55                      ;                                       9581
21633 075134  000207                            .WORD   207
21634 075136  010526                            .WORD   SYNC
21635 075140  024060                            .WORD   DUMPER
21636 075142  012746  007472                    MOV     #PHR.5,-(SP)            ;                                       9582
21637 075146  012746  007330                    MOV     #WRD.75,-(SP)
21638 075152  012746  007316                    MOV     #WRD.74,-(SP)
21639 075156  012746  007174                    MOV     #WRD.61,-(SP)
21640 075162  012746  006046                    MOV     #FOR.FMT,-(SP)
21641 075166  012746  000005                    MOV     #5,-(SP)
21642 075172  010600                            MOV     SP,R0                   ; SP,*
21643 075174  104414                            TRAP    14
21644 075176  010316                            MOV     R3,(SP)                 ; CHANNEL,*                             9583
21645 075200  010446                            MOV     R4,-(SP)                ; PLOG,*
21646 075202  012746  005534                    MOV     #FMT.21,-(SP)
21647 075206  012746  000003                    MOV     #3,-(SP)
21648 075212  010600                            MOV     SP,R0                   ; SP,*
```

```
21650                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
21651                                   ;                                                       14-May-1981 13:05:08    PA:<
21652
21653 075214  104414                            TRAP    14
21654 075216  062706  000022                      ADD     #22,SP                ;                                         9580
21655 075222  005201                    8$:     INC     R1                    ; WRD.CNT                                 9576
21656 075224  020127  000014                      CMP     R1,#14                ; WRD.CNT,*
21657 075230  003733                            BLE     7$
21658 075232  022626                            CMP     (SP)+,(SP)+           ;                                         9554
21659 075234  104467                            TRAP    67                    ;                                         9584
21660 075236  006000                            ROR     R0
21661 075240  103620                            BLO     3$
21662 075242  005203                            INC     R3                    ; CHANNEL                                 9553
21663 075244  020327  000043                      CMP     R3,#43                ; CHANNEL,*
21664 075250  003614                            BLE     3$
21665 075252  005204                            INC     R4                    ; PLOG                                    9550
21666 075254  020427  000005                      CMP     R4,#5                 ; PLOG,*
21667 075260  003607                            BLE     2$
21668 075262  012705  177777                      MOV     #-1,R5                ; *,BITS.XFERED                           9591
21669 075266  005003                            CLR     R3                    ; CHANNEL                                 9593
21670 075270  104402                    9$:     TRAP    2                     ;                                         9594
21671 075272  152777  000040  115714              BISB    #40,@ML.REG+40        ;                                         9595
21672 075300  016704  116276                      MOV     ML.DUT,R4
21673 075304  042704  177770                      BIC     #177770,R4
21674 075310  142777  000007  115676              BICB    #7,@ML.REG+40
21675 075316  150477  115672                      BISB    R4,@ML.REG+40
21676 075322  005205                            INC     R5                    ; BITS.XFERED                             9597
21677 075324  005001                            CLR     R1                    ; CNT                                     9599
21678 075326  010102                    10$:    MOV     R1,R2                 ; CNT,*                                   9600
21679 075330  006302                            ASL     R2
21680 075332  005062  011600                      CLR     IO.BUF(R2)
21681 075336  005201                            INC     R1                    ; CNT                                     9599
21682 075340  020127  000024                      CMP     R1,#24                ; CNT,*
21683 075344  003770                            BLE     10$
21684 075346  012746  000072                      MOV     #72,-(SP)             ;                                         9602
21685 075352  010346                            MOV     R3,-(SP)              ; CHANNEL,*
21686 075354  004767  120222                      JSR     PC,FORCE.REM
21687 075360  004767  120504                      JSR     PC,GD.BLK.XFER        ;                                         9603
21688 075364  142777  000002  115702              BICB    #2,@ML.REG+120        ;                                         9604
21689 075372  012777  000071  115554              MOV     #71,@ML.REG           ;                                         9605
21690 075400  105777  115620            11$:     TSTB    @ML.REG+50
21691 075404  100375                            BPL     11$
21692 075406  005002                            CLR     R2                    ; BIT.OFFSET                              9608
21693 075410  010246                    12$:    MOV     R2,-(SP)              ; BIT.OFFSET,*                            9610
21694 075412  060516                            ADD     R5,(SP)               ; BITS.XFERED,*
21695 075414  004767  120010                      JSR     PC,FIND.COMP.BIT
21696 075420  005726                            TST     (SP)+
21697 075422  006000                            ROR     R0
21698 075424  103433                            BLO     13$
21699 075426  104455                            TRAP    55                    ;                                         9613
21700 075430  000206                            .WORD   206
21701 075432  010526                            .WORD   SYNC
21702 075434  024060                            .WORD   DUMPER
21703 075436  012746  007366                      MOV     #PHR.1,-(SP)          ;                                         9614
21704 075442  012746  007330                      MOV     #WRD.75,-(SP)
```

```
21706                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
21707                                   ;                                                       14-May-1981 13:05:08    PA:<
21708
21709 075446  012746  007316                     MOV     #WRD.74,-(SP)
21710 075452  012746  006034                     MOV     #THR.FMT,-(SP)
21711 075456  012746  000004                     MOV     #4,-(SP)
21712 075462  010600                             MOV     SP,R0                   ; SP,*
21713 075464  104414                             TRAP    14
21714 075466  010316                             MOV     R3,(SP)                 ; CHANNEL,*                             9615
21715 075470  012746  000072                     MOV     #72,-(SP)
21716 075474  012746  005534                     MOV     #FMT.21,-(SP)
21717 075500  012746  000003                     MOV     #3,-(SP)
21718 075504  010600                             MOV     SP,R0                   ; SP,*
21719 075506  104414                             TRAP    14
21720 075510  062706  000020                     ADD     #20,SP                  ;                                       9612
21721 075514  062702  000044           13$:      ADD     #44,R2                  ; *,BIT.OFFSET                          9608
21722 075520  020227  000264                     CMP     R2,#264                 ; BIT.OFFSET,*
21723 075524  003731                             BLE     12$
21724 075526  005001                             CLR     R1                      ; WRD.CNT                               9618
21725 075530  010102           14$:      MOV     R1,R2                   ; WRD.CNT,*                             9620
21726 075532  006302                             ASL     R2
21727 075534  005762  011600                     TST     IO.BUF(R2)
21728 075540  001437                             BEQ     15$
21729 075542  104455                             TRAP    55                      ;                                       9623
21730 075544  000207                             .WORD   207
21731 075546  010526                             .WORD   SYNC
21732 075550  024060                             .WORD   DUMPER
21733 075552  012746  007240                     MOV     #WRD.67,-(SP)           ;                                       9624
21734 075556  012746  006426                     MOV     #WRD.12,-(SP)
21735 075562  012746  007330                     MOV     #WRD.75,-(SP)
21736 075566  012746  007316                     MOV     #WRD.74,-(SP)
21737 075572  012746  007174                     MOV     #WRD.61,-(SP)
21738 075576  012746  006062                     MOV     #FIV.FMT,-(SP)
21739 075602  012746  000006                     MOV     #6,-(SP)
21740 075606  010600                             MOV     SP,R0                   ; SP,*
21741 075610  104414                             TRAP    14
21742 075612  010316                             MOV     R3,(SP)                 ; CHANNEL,*                             9625
21743 075614  012746  000072                     MOV     #72,-(SP)
21744 075620  012746  005534                     MOV     #FMT.21,-(SP)
21745 075624  012746  000003                     MOV     #3,-(SP)
21746 075630  010600                             MOV     SP,R0                   ; SP,*
21747 075632  104414                             TRAP    14
21748 075634  062706  000024                     ADD     #24,SP                  ;                                       9622
21749 075640  005201           15$:      INC     R1                      ; WRD.CNT                               9618
21750 075642  020127  000014                     CMP     R1,#14                  ; WRD.CNT,*
21751 075646  003730                             BLE     14$
21752 075650  022626                             CMP     (SP)+,(SP)+             ;                                       9594
21753 075652  104467                             TRAP    67                      ;                                       9626
21754 075654  006000                             ROR     R0
21755 075656  103604                             BLO     9$
21756 075660  005203                             INC     R3                      ; CHANNEL                               9593
21757 075662  020327  000043                     CMP     R3,#43                  ; CHANNEL,*
21758 075666  003600                             BLE     9$
21759 075670  000207                             RTS     PC                      ;                                       9507
21760
```

```
21762                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
21763                                   ;                                                       14-May-1981 13:05:08    PA:<
21764
21765                                   ; Routine Size:  271 words
21766                                   ; Maximum stack depth per invocation:  18 words
21771
21772
21776
21777                                           .SBTTL  T51
21781 075672                            T51::
21782 075672  004767  176736            1$:     JSR     PC,$T51                         ;       9629
21783 075676  104466                            TRAP    66
21784 075700  006000                            ROR     R0
21785 075702  103773                            BLO     1$
21786 075704  000207                            RTS     PC
21787
21788                                   ; Routine Size:  6 words
21789                                   ; Maximum stack depth per invocation:  0 words
21794
21795
21796 ;         9632  !<BLF/PAGE>
```

```
21798 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
21799 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (102)
21800
21801 ;          9633  !
21802 ;          9634  BGNTST;
21803 ;          9635
21804 ;          9636  !++
21805 ;          9637  ! TEST NUMBER:  TST 52
21806 ;          9638  !
21807 ;          9639  ! TEST NAME:            UNCORRECTABLE ERROR SYNDRONE DECODE TEST AT CHANNELS > 35
21808 ;          9640  !
21809 ;          9641  ! TEST DESCRIPTION:
21810 ;          9642  !       TEST SYNDRONE DECODE TO DETECT BUT
21811 ;          9643  !       NOT CORRECT UNCORRECTABLE CHANNEL
21812 ;          9644  !       ERRORS BY:
21813 ;          9645  !
21814 ;          9646  !       1.  VIA ECC DIAG MODE FORCE SYNDRONE
21815 ;          9647  !           BITS TO INDICATE CHANNEL ERRORS
21816 ;          9648  !           AT WORDS OF A CRC GROUP BUT AT CHANNELS
21817 ;          9649  !           GREATER THAN 35.
21818 ;          9650  !
21819 ;          9651  !       2.  DO A MASS BUS READ TRANSFER
21820 ;          9652  !
21821 ;          9653  !       3.  THEN EXAMIN THE IO BUF FOR NO
21822 ;          9654  !           BITS COMPLIMENTED AND ECH  AND
21823 ;          9655  !           BITS SET
21824 ;          9656  !
21825 ;          9657  ! IMPLICIT INPUTS:
21826 ;          9658  !       IO_BUF
21827 ;          9659  !       A  VECTOR OF 256 WORDS WHERE DATA
21828 ;          9660  !       FOR MBUS READS AND WRITES TRANSFERS
21829 ;          9661  !       IS FOUND.
21830 ;          9662  !
21831 ;          9663  !
21832 ;          9664  !--
21833 ;          9665
21834 ;          9666  local
21835 ;          9667      PLOG;                                !INDEX INTO REMAINDER TABLE 'REM_TBL'
21836 ;          9668
21837 ;          9669  BAI = ONE;                               !FIRST LETS CLEAR THE ML11'S GOOD BLOCK
21838 ;          9670  IO_BUF = ZEROES;                         !BY WRITING ZEROES TO IT
21839 ;          9671  GD_BLK_XFER ();
21840 ;          9672  ML[S1 = write;
21841 ;          9673  TIME_OUT_LOOP;
21842 ;          9674
21843 ;          9675  incr PLOG_SEL from 0 to 6 do             !TEST SYNDRONE DECODE AT 7 DIFFERENT PLOG VALUES
21844 ;          9676      begin
21845 ;          9677
21846 ;          9678      case .PLOG_SEL from 0 to 6 of        !SELECT A PLOG VALUE
21847 ;          9679          set
21848 ;          9680
21849 ;          9681          [0] :                            !POINTS TO WORD 0 IN THE CRC GROUP
21850 ;          9682              PLOG = 0;
21851 ;          9683
21852 ;          9684          [1] :                            !POINTS TO WORD 1 IN THE CRC GROUP
21853 ;ML4                                                     14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
21854 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (102)
```

```
21855
21856 ;          9685              PLOG = 1;
21857 ;          9686
21858 ;          9687          [2] :                                   !POINTS TO WORD 2 IN THE CRC GROUP
21859 ;          9688              PLOG = 2;
21860 ;          9689
21861 ;          9690          [3] :                                   !POINTS TO WORD 3 IN THE CRC GROUP
21862 ;          9691              PLOG = 3;
21863 ;          9692
21864 ;          9693          [4] :                                   !POINTS TO WORD 4 IN THE CRC GROUP
21865 ;          9694              PLOG = 4;
21866 ;          9695
21867 ;          9696          [5] :                                   !POINTS TO WORD 5 IN THE CRC GROUP
21868 ;          9697              PLOG = 5;
21869 ;          9698
21870 ;          9699          [6] :                                   !POINTS TO ALL 6 WORDS IN THE CRC GROUP
21871 ;          9700              PLOG = 58;
21872 ;          9701          tes;
21873 ;          9702
21874 ;          9703      incr CHANNEL from 36 to 62 do               !TEST AT CHANNELS > 35 TO FORCE UNCORRECTABLE ERROR
21875 ;          9704          begin
21876 ;          9705          BGNSUB;                                 !START OF SCOPE LOOP
21877 ;          9706          CLR_MBUS;                               !CLEAR THE DRIVE
21878 ;          9707
21879 ;          9708          incr COUNT from 0 to 255 do             !CLEAR THE FIRST CRC GROUP IN THE IO_BUF
21880 ;          9709              IO_BUF [.COUNT] = ZEROES;
21881 ;          9710
21882 ;          9711          FORCE_REM (.PLOG, .CHANNEL);            !FORCE UNCORRECTABLE ERROR AT THIS WORD AND CHANNEL
21883 ;          9712          GD_BLK_XFER ();                         !SET UP A GOOD BLOCK TRANSFER
21884 ;          9713          ECC_DIS = ZERO;                         !CLEAR ECC DISABLE
21885 ;          9714          MLCS1 = read;                           !DO A READ TRANSFER
21886 ;          9715          TIME_OUT_LOOP;                          !WAIT FOR THE  TRANSFER TO COMPLETE
21887 ;          9716
21888 ;          9717          if not ((.ECH_ERR) and (.UNC_ERR))      !SEE IF THE ERROR WAS DETECTED
21889 ;          9718          then
21890 ;          9719              begin                               !ERROR IF NOT DECTECTED
21891 ;          9720              ERRDF (136, SYNC, DUMPER);
21892 ;          9721              PRINTB (FOR_FMT, WRD_67, WRD_10, WRD_76, WRD_9);
21893 ;          9722              PRINTB (FMT_21, .PLOG, .CHANNEL);
21894 ;          9723              end;
21895 ;          9724
21896 ;          9725          incr CNT from 0 to 12 do                !SEE IF ANY CORRECTION WAS DONE
21897 ;          9726
21898 ;          9727              if .IO_BUF [.CNT] neq ZEROES        !SEARCH THE IO_BUF FOR ALL ZEROES
21899 ;          9728              then
21900 ;          9729                  begin                           !REPORT AN ERROR IF ANY CORRECTION WAS DONE
21901 ;          9730                  ERRDF (137, SYNC, DUMPER);
21902 ;          9731                  PRINTB (FIV_FMT, WRD_61, WRD_74, WRD_75, WRD_12, WRD_67);
21903 ;          9732                  PRINTB (FMT_21, .PLOG, .CHANNEL);
21904 ;          9733                  end;
21905 ;          9734
21906 ;          9735          ENDSUB;                                 !END OF SCOPE LOOP
21907 ;          9736          end;
21908 ;ML4                                                            14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
21909 ;                                                              14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (102)
21910
21911 ;          9737
```

```
21912 ;          9738      end;
21913 ;          9739
21914 ;          9740  ENDTST;
21918
21919                                        .SBTTL  $T52
21923 075706  004167  106204          $T52:  JSR     R1,$SAVE5                ;                                  9631
21924 075712  152777  000010  115274         BISB    #10,@ML.REG+40           ;                                  9669
21925 075720  005067  113654                 CLR     IO.BUF                   ;                                  9670
21926 075724  004767  120140                 JSR     PC,GD.BLK.XFER           ;                                  9671
21927 075730  012777  000061  115216         MOV     #61,@ML.REG              ;                                  9672
21928 075736  105777  115262           1$:   TSTB    @ML.REG+50
21929 075742  100375                        BPL     1$
21930 075744  005005                        CLR     R5                       ; PLOG.SEL                         9675
21931 075746  010504                  2$:   MOV     R5,R4                    ; PLOG.SEL,*                       9678
21932 075750  006304                        ASL     R4
21933 075752  066407  075756               ADD     3$(R4),PC
21934 075756  000016                  3$:   .WORD   4$-3$
21935 075760  000022                        .WORD   5$-3$
21936 075762  000030                        .WORD   6$-3$
21937 075764  000036                        .WORD   7$-3$
21938 075766  000044                        .WORD   8$-3$
21939 075770  000052                        .WORD   9$-3$
21940 075772  000060                        .WORD   10$-3$
21941 075774  005003                  4$:   CLR     R3                       ; PLOG                             9682
21942 075776  000421                        BR      11$                      ;                                  9678
21943 076000  012703  000001           5$:   MOV     #1,R3                    ; *,PLOG                           9685
21944 076004  000416                        BR      11$                      ;                                  9678
21945 076006  012703  000002           6$:   MOV     #2,R3                    ; *,PLOG                           9688
21946 076012  000413                        BR      11$                      ;                                  9678
21947 076014  012703  000003           7$:   MOV     #3,R3                    ; *,PLOG                           9691
21948 076020  000410                        BR      11$                      ;                                  9678
21949 076022  012703  000004           8$:   MOV     #4,R3                    ; *,PLOG                           9694
21950 076026  000405                        BR      11$                      ;                                  9678
21951 076030  012703  000005           9$:   MOV     #5,R3                    ; *,PLOG                           9697
21952 076034  000402                        BR      11$                      ;                                  9678
21953 076036  012703  000072          10$:   MOV     #72,R3                   ; *,PLOG                           9700
21954 076042  012704  000044          11$:   MOV     #44,R4                   ; *,CHANNEL                        9703
21955 076046  104402                  12$:   TRAP    2                        ;                                  9704
21956 076050  152777  000040  115136         BISB    #40,@ML.REG+40           ;                                  9705
21957 076056  016702  115520                 MOV     ML.DUT,R2
21958 076062  042702  177770                 BIC     #177770,R2
21959 076066  142777  000007  115120         BICB    #7,@ML.REG+40
21960 076074  150277  115114                 BISB    R2,@ML.REG+40
21961 076100  005001                        CLR     R1                       ; COUNT                            9708
21962 076102  010102                  13$:  MOV     R1,R2                    ; COUNT,*                          9709
21963                                 ;ML4                                                 14-May-1981 13:48:54  TOPS
21964                                 ;                                                    14-May-1981 13:05:08  PA:<
21965
21966 076104  006302                        ASL     R2
21967 076106  005062  011600               CLR     IO.BUF(R2)
21968 076112  005201                        INC     R1                       ; COUNT                            9708
21969 076114  020127  000377               CMP     R1,#377                  ; COUNT,*
21970 076120  003770                        BLE     13$
21971 076122  010346                        MOV     R3,-(SP)                 ; PLOG,*                           9711
21972 076124  010446                        MOV     R4,-(SP)                 ; CHANNEL,*
21973 076126  004767  117450               JSR     PC,FORCE.REM
21974 076132  004767  117732               JSR     PC,GD.BLK.XFER           ;                                  9712
```

```
 21975 076136  142777  000002  115130          BICB    #2,@ML.REG+120          ;                                       9713
 21976 076144  012777  000071  115002          MOV     #71,@ML.REG             ;                                       9714
 21977 076152  105777  115046          14$:    TSTB    @ML.REG+50
 21978 076156  100375                          BPL     14$
 21979 076160  132777  000100  115046          BITB    #100,@ML.REG+60         ;                                       9717
 21980 076166  001403                          BEQ     15$
 21981 076170  005777  115170                  TST     @ML.REG+210
 21982 076174  100434                          BMI     16$
 21983 076176  104455                  15$:    TRAP    55                      ;                                       9720
 21984 076200  000210                          .WORD   210
 21985 076202  010526                          .WORD   SYNC
 21986 076204  024060                          .WORD   DUMPER
 21987 076206  012746  006374                  MOV     #WRD.9,-(SP)            ;                                       9721
 21988 076212  012746  007340                  MOV     #WRD.76,-(SP)
 21989 076216  012746  006406                  MOV     #WRD.10,-(SP)
 21990 076222  012746  007240                  MOV     #WRD.67,-(SP)
 21991 076226  012746  006046                  MOV     #FOR.FMT,-(SP)
 21992 076232  012746  000005                  MOV     #5,-(SP)
 21993 076236  010600                          MOV     SP,R0                   ; SP,*
 21994 076240  104414                          TRAP    14
 21995 076242  010416                          MOV     R4,(SP)                 ; CHANNEL,*                             9722
 21996 076244  010346                          MOV     R3,-(SP)                ; PLOG,*
 21997 076246  012746  005534                  MOV     #FMT.21,-(SP)
 21998 076252  012746  000003                  MOV     #3,-(SP)
 21999 076256  010600                          MOV     SP,R0                   ; SP,*
 22000 076260  104414                          TRAP    14
 22001 076262  062706  000022                  ADD     #22,SP                  ;                                       9719
 22002 076266  005001                  16$:    CLR     R1                      ; CNT                                   9725
 22003 076270  010102                  17$:    MOV     R1,R2                   ; CNT,*                                 9727
 22004 076272  006302                          ASL     R2
 22005 076274  005762  011600                  TST     IO.BUF(R2)
 22006 076300  001436                          BEQ     18$
 22007 076302  104455                          TRAP    55                      ;                                       9730
 22008 076304  000211                          .WORD   211
 22009 076306  010526                          .WORD   SYNC
 22010 076310  024060                          .WORD   DUMPER
 22011 076312  012746  007240                  MOV     #WRD.67,-(SP)           ;                                       9731
 22012 076316  012746  006426                  MOV     #WRD.12,-(SP)
 22013 076322  012746  007330                  MOV     #WRD.75,-(SP)
 22014 076326  012746  007316                  MOV     #WRD.74,-(SP)
 22015 076332  012746  007174                  MOV     #WRD.61,-(SP)
 22016 076336  012746  006062                  MOV     #FIV.FMT,-(SP)
 22017 076342  012746  000006                  MOV     #6,-(SP)
 22018                         ;ML4                                    14-May-1981 13:48:54    TOPS
 22019                         ;                                       14-May-1981 13:05:08    PA:<
 22020
 22021 076346  010600                          MOV     SP,R0                   ; SP,*
 22022 076350  104414                          TRAP    14
 22023 076352  010416                          MOV     R4,(SP)                 ; CHANNEL,*                             9732
 22024 076354  010346                          MOV     R3,-(SP)                ; PLOG,*
 22025 076356  012746  005534                  MOV     #FMT.21,-(SP)
 22026 076362  012746  000003                  MOV     #3,-(SP)
 22027 076366  010600                          MOV     SP,R0                   ; SP,*
 22028 076370  104414                          TRAP    14
 22029 076372  062706  000024                  ADD     #24,SP                  ;                                       9729
 22030 076376  005201                  18$:    INC     R1                      ; CNT                                   9725
 22031 076400  020127  000014                  CMP     R1,#14                  ; CNT,*
```

```
 22032 076404  003731                    BLE     17$
 22033 076406  022626                    CMP     (SP)+,(SP)+                     ;                       9704
 22034 076410  104467                    TRAP    67                              ;                       9733
 22035 076412  006000                    ROR     R0
 22036 076414  103614                    BLO     12$
 22037 076416  005204                    INC     R4                              ; CHANNEL               9703
 22038 076420  020427  000076            CMP     R4,#76                          ; CHANNEL,*
 22039 076424  003610                    BLE     12$
 22040 076426  005205                    INC     R5                              ; PLOG.SEL              9675
 22041 076430  020527  000006            CMP     R5,#6                           ; PLOG.SEL,*
 22042 076434  003002                    BGT     19$
 22043 076436  000167  177304            JMP     2$
 22044 076442  000207              19$:  RTS     PC                              ;                       9631
 22045
 22046                                  ; Routine Size:  175 words
 22047                                  ; Maximum stack depth per invocation:  18 words
 22052
 22053
 22057
 22058                                          .SBTTL  T52
 22062 076444                           T52::
 22063 076444  004767  177236           1$:    JSR     PC,$T52                         ;                       9738
 22064 076450  104466                    TRAP    66
 22065 076452  006000                    ROR     R0
 22066 076454  103773                    BLO     1$
 22067 076456  000207                    RTS     PC
 22068
 22069                                  ; Routine Size:  6 words
 22070                                  ; Maximum stack depth per invocation:  0 words
 22078
 22079
 22080 ;          9741   !<BLF/PAGE>
```

```
22082 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
22083 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (103)
22084
22085 ;          9742  !
22086 ;          9743  BGNTST;
22087 ;          9744
22088 ;          9745  !++
22089 ;          9746  ! TEST NUMBER:  TST 53
22090 ;          9747  !
22091 ;          9748  ! TEST NAME:            MULTIPLE CHANNEL ERROR SYNDRONE DECODE TEST
22092 ;          9749  !
22093 ;          9750  ! TEST DESCRIPTION:
22094 ;          9751  !       TEST SYNDRONE DECODE TO DETECT BUT NOT
22095 ;          9752  !       CORRECT UNCORRECTABLE MULTIPLE CHANNEL
22096 ;          9753  !       ERRORS BY:
22097 ;          9754  !
22098 ;          9755  !       1.  VIA ECC DIAGNOSTIC MODE FORCE SYNDRONE
22099 ;          9756  !           BITS TO INDICATE MULTIPLE CHANNEL ERRORS
22100 ;          9757  !           BY ASSIGNING CRC_A EQUAL  ) CRC_B
22101 ;          9758  !
22102 ;          9759  !       2.  DO A MASS BUS READ TRANSFER
22103 ;          9760  !
22104 ;          9761  !       3.  THEN EXAMIN THE IO_BUF FOR  NO BITS
22105 ;          9762  !           COMPLIMENTED AND ECH AND UNC BITS SET
22106 ;          9763  !
22107 ;          9764  ! IMPLICIT INPUTS:
22108 ;          9765  !       IO_BUF
22109 ;          9766  !       A  VECTOR OF 256 WORDS WHERE DATA
22110 ;          9767  !       FOR MBUS READS AND WRITES TRANSFERS
22111 ;          9768  !       IS FOUND.
22112 ;          9769  !
22113 ;          9770  !
22114 ;          9771  !--
22115 ;          9772
22116 ;          9773  local
22117 ;          9774      PLOG;                             !INDEX INTO REMAINDER TABLE 'REM_TBL'
22118 ;          9775
22119 ;          9776  BAI = ONE;                            !FIRST LETS CLEAR THE ML11'S GOOD
22120 ;          9777  IO_BUF = ZEROES;                      !BLOCK BY WRITING ZEROES TO IT
22121 ;          9778  GD_BLK_XFER ();
22122 ;          9779  ML[S1 = write;
22123 ;          9780  TIME_OUT_LOOP;
22124 ;          9781
22125 ;          9782  incr PLOG_SEL from 0 to 6 do          !TEST SYNDRONE DECODE AT 7 DIFFERENT PLOG VALUES
22126 ;          9783      begin
22127 ;          9784
22128 ;          9785      case .PLOG_SEL from 0 to 6 of     !SELECT A PLOG VALUE
22129 ;          9786          set
22130 ;          9787
22131 ;          9788          [0] :                         !POINTS TO CRC WORD 0
22132 ;          9789              PLOG = 0;
22133 ;          9790
22134 ;          9791          [1] :                         !POINTS TO CRC WORD 1
22135 ;          9792              PLOG = 1;
22136 ;          9793
22137 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
22138 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (103)
```

```
22139
22140 ;          9794          [2] :                                   !POINTS TO CRC WORD 2
22141 ;          9795              PLOG = 2;
22142 ;          9796
22143 ;          9797          [3] :                                   !POINTS TO CRC WORD 3
22144 ;          9798              PLOG = 3;
22145 ;          9799
22146 ;          9800          [4] :                                   !POINTS TO CRC WORD 4
22147 ;          9801              PLOG = 4;
22148 ;          9802
22149 ;          9803          [5] :                                   !POINTS TO CRC WORD 5
22150 ;          9804              PLOG = 5;
22151 ;          9805
22152 ;          9806          [6] :                                   !POINTS TO ALL 6 WORDS IN CRC GROUP
22153 ;          9807              PLOG = 58;
22154 ;          9808          tes;
22155 ;          9809
22156 ;          9810      incr CHANNEL from 1 to 35 do                !TEST 35 CHANNELS STARTING AT CHANNEL 1
22157 ;          9811          begin
22158 ;          9812          BGNSUB;                                 !START OF SCOPE LOOP
22159 ;          9813          CLR_MBUS;                               !CLEAR THE DRIVE
22160 ;          9814
22161 ;          9815          incr COUNT from 0 to 255 do             !CLEAR FIRST CRC GROUP IN THE IO_BUF
22162 ;          9816              IO_BUF [.COUNT] = ZEROES;
22163 ;          9817
22164 ;          9818          FORCE_REM (.PLOG, .CHANNEL);            !FORCE ERR AT THIS WORD AND CHANNEL
22165 ;          9819          CRC_A = .CRC_B;                         !CRC_A EQL TO CRC_B CAUSES THE UNC ERROR
22166 ;          9820          GD_BLK_XFER ();                         !SET UP A GOOD BLOCK TRANSFER
22167 ;          9821          ECC_DIS = ZERO;                         !CLEAR ECC DISABLE
22168 ;          9822          MLCS1 = read;                           !DO A READ TRANSFER
22169 ;          9823          TIME_OUT_LOOP;                          !WAIR UNTIL THE TRANSFER IS COMPLETE
22170 ;          9824
22171 ;          9825          if not ((.ECH_ERR) and (.UNC_ERR))      !SEE IF THE ERROR WAS DETECTED
22172 ;          9826          then
22173 ;          9827              begin                               !REPORT ERROR IF NOT DETECTED
22174 ;          9828              ERRDF (138, SYNC, DUMPER);
22175 ;          9829              PRINTB (FOR_FMT, WRD_77, WRD_10, WRD_76, WRD_9);
22176 ;          9830              PRINTB (FMT_21, .PLOG, .CHANNEL);
22177 ;          9831              end;
22178 ;          9832
22179 ;          9833          incr CNT from 0 to 12 do                !SEE IF ANY ERROR CORRECTION WAS DONE
22180 ;          9834
22181 ;          9835              if .IO_BUF [.CNT] neq ZEROES        !TEST THE IO_BUF FOR ALL ZEROES
22182 ;          9836              then
22183 ;          9837                  begin                           !ERROR IF ANY BITS GOT FLIPPED
22184 ;          9838                  ERRDF (139, SYNC, DUMPER);
22185 ;          9839                  PRINTB (FIV_FMT, WRD_61, WRD_74, WRD_75, WRD_12, WRD_67);
22186 ;          9840                  PRINTB (FMT_21, .PLOG, .CHANNEL);
22187 ;          9841                  end;
22188 ;          9842
22189 ;          9843          ENDSUB;                                 !END OF SCOPE LOOP
22190 ;          9844          end;
22191 ;          9845
22192 ;ML4                                                            14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
22193 ;                                                               14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (103)
22194
22195 ;          9846      end;
```

```
22196 ;          9847
22197 ;          9848  ENDTST:
22201
22202                                          .SBTTL  $T53
22206 076460  004167  105432          $T53:   JSR     R1,$SAVE5                ;                              9740
22207 076464  152777  000010  114522          BISB    #10,@ML.REG+40           ;                              9776
22208 076472  005067  113102                  CLR     IO.BUF                   ;                              9777
22209 076476  004767  117366                  JSR     PC,GD.BLK.XFER           ;                              9778
22210 076502  012777  000061  114444          MOV     #61,@ML.REG              ;                              9779
22211 076510  105777  114510          1$:     TSTB    @ML.REG+50
22212 076514  100375                          BPL     1$
22213 076516  005005                          CLR     R5                       ; PLOG.SEL                     9782
22214 076520  010504                  2$:     MOV     R5,R4                    ; PLOG.SEL,*                   9785
22215 076522  006304                          ASL     R4
22216 076524  066407  076530                  ADD     3$(R4),PC
22217 076530  000016                  3$:     .WORD   4$-3$
22218 076532  000022                          .WORD   5$-3$
22219 076534  000030                          .WORD   6$-3$
22220 076536  000036                          .WORD   7$-3$
22221 076540  000044                          .WORD   8$-3$
22222 076542  000052                          .WORD   9$-3$
22223 076544  000060                          .WORD   10$-3$
22224 076546  005003                  4$:     CLR     R3                       ; PLOG                         9789
22225 076550  000421                          BR      11$                      ;                              9785
22226 076552  012703  000001          5$:     MOV     #1,R3                    ; *,PLOG                       9792
22227 076556  000416                          BR      11$                      ;                              9785
22228 076560  012703  000002          6$:     MOV     #2,R3                    ; *,PLOG                       9795
22229 076564  000413                          BR      11$                      ;                              9785
22230 076566  012703  000003          7$:     MOV     #3,R3                    ; *,PLOG                       9798
22231 076572  000410                          BR      11$                      ;                              9785
22232 076574  012703  000004          8$:     MOV     #4,R3                    ; *,PLOG                       9801
22233 076600  000405                          BR      11$                      ;                              9785
22234 076602  012703  000005          9$:     MOV     #5,R3                    ; *,PLOG                       9804
22235 076606  000402                          BR      11$                      ;                              9785
22236 076610  012703  000072          10$:    MOV     #72,R3                   ; *,PLOG                       9807
22237 076614  012704  000001          11$:    MOV     #1,R4                    ; *,CHANNEL                    9810
22238 076620  104402                  12$:    TRAP    2                        ;                              9811
22239 076622  152777  000040  114364          BISB    #40,@ML.REG+40           ;                              9812
22240 076630  016702  114746                  MOV     ML.DUT,R2
22241 076634  042702  177770                  BIC     #177770,R2
22242 076640  142777  000007  114346          BICB    #7,@ML.REG+40
22243 076646  150277  114342                  BISB    R2,@ML.REG+40
22244 076652  005001                          CLR     R1                       ; COUNT                        9815
22245 076654  010102                  13$:    MOV     R1,R2                    ; COUNT,*                      9816
22246 076656  006302                          ASL     R2
22247                                  ;ML4                                                    14-May-1981 13:48:54   TOPS
22248                                  ;                                                       14-May-1981 13:05:08   PA:<
22249
22250 076660  005062  011600                  CL      IO.BUF(R2)
22251 076664  005201                          INC     R1                       ; COUNT                        9815
22252 076666  020127  000377                  CMP     R1,#377                  ; COUNT,*
22253 076672  003770                          BLE     13$
22254 076674  010346                          MOV     R3,-(SP)                 ; PLOG,*                       9818
22255 076676  010446                          MOV     R4,-(SP)                 ; CHANNEL,*
22256 076700  004767  116676                  JSR     PC,FORCE.REM
22257 076704  117702  114424                  MOVB    @ML.REG+160,R2           ;                              9819
22258 076710  042702  177700                  BIC     #177700,R2
```

```
22259 076714  142777  000077  114402          BICB    #77,@ML.REG+150
22260 076722  150277  114376                  BISB    R2,@ML.REG+150
22261 076726  004767  117136                  JSR     PC,GD.BLK.XFER          ;                       9820
22262 076732  142777  000002  114334          BICB    #2,@ML.REG+120          ;                       9821
22263 076740  012777  000071  114206          MOV     #71,@ML REG             ;                       9822
22264 076746  105777  114252          14$:    TSTB    @ML.REG+50
22265 076752  100375                          BPL     14$
22266 076754  132777  000100  114252          BITB    #100,@ML.REG+60         ;                       9825
22257 076762  001403                          BEQ     15$
22268 076764  005777  114374                  TST     @ML.REG+210
22269 076770  100434                          BMI     16$
22270 076772  104455                  15$:    TRAP    55                      ;                       9828
22271 076774  000212                          .WORD   212
22272 076776  010526                          .WORD   SYNC
22273 077000  024060                          .WORD   DUMPER
22274 077002  012746  006374                  MOV     #WRD.9,-(SP)            ;                       9829
22275 077006  012746  007340                  MOV     #WRD.76,-(SP)
22276 077012  012746  006406                  MOV     #WRD.10,-(SP)
22277 077016  012746  007346                  MOV     #WRD.77,-(SP)
22278 077022  012746  006046                  MOV     #FOR.FMT,-(SP)
22279 077026  012746  000005                  MOV     #5,-(SP)
22280 077032  010600                          MOV     SP,R0                   ; SP,*
22281 077034  104414                          TRAP    14
22282 077036  010416                          MOV     R4,(SP)                 ; CHANNEL,*             9830
22283 077040  010346                          MOV     R3,-(SP)                ; PLOG,*
22284 077042  012746  005534                  MOV     #FMT.21,-(SP)
22285 077046  012746  000003                  MOV     #3,-(SP)
22286 077052  010600                          MOV     SP,R0                   ; SP,*
22287 077054  104414                          TRAP    14
22288 077056  062706  000022                  ADD     #22,SP                  ;                       9827
22289 077062  005001                  16$:    CLR     R1                      ; CNT                   9833
22290 077064  010102                  17$:    MOV     R1,R2                   ; CNT,*                 9835
22291 077066  006302                          ASL     R2
22292 077070  005762  011600                  TST     IO.BUF(R2)
22293 077074  001436                          BEQ     18$
22294 077076  104455                          TRAP    55                      ;                       9838
22295 077100  000213                          .WORD   213
22296 077102  010526                          .WORD   SYNC
22297 077104  024060                          .WORD   DUMPER
22298 077106  012746  007240                  MOV     #WRD.67,-(SP)           ;                       9839
22299 077112  012746  006426                  MOV     #WRD.12,-(SP)
22300 077116  012746  007330                  MOV     #WRD.75,-(SP)
22301 077122  012746  007316                  MOV     #WRD.74,-(SP)
22302                         ;ML4                                    14-May-1981 13:48:54    TOPS
22303                         ;                                       14-May-1981 13:05:08    PA:<
22304
22305 077126  012746  007174                  MOV     #WRD.61,-(SP)
22306 077132  012746  006062                  MOV     #FIV.FMT,-(SP)
22307 077136  012746  000006                  MOV     #6,-(SP)
22308 077142  010600                          MOV     SP,R0                   ; SP,*
22309 077144  104414                          TRAP    14
22310 077146  010416                          MOV     R4,(SP)                 ; CHANNEL,*             9840
22311 077150  010346                          MOV     R3,-(SP)                ; PLOG,*
22312 077152  012746  005534                  MOV     #FMT.21,-(SP)
22313 077156  012746  000003                  MOV     #3,-(SP)
22314 077162  010600                          MOV     SP,R0                   ; SP,*
22315 077164  104414                          TRAP    14
```

```
 22316 077166  062706  000024            ADD     #24,SP                          ;                                                  9837
 22317 077172  005201                18$:  INC     R1                              ; CNT                                              9833
 22318 077174  020127  000014            CMP     R1,#14                          ; CNT,*
 22319 077200  003731                    BLE     17$
 22320 077202  022626                    CMP     (SP)+,(SP)+                     ;                                                  9811
 22321 077204  104467                    TRAP    67                              ;                                                  9841
 22322 077206  006000                    ROR     R0
 22323 077210  103603                    BLO     12$
 22324 077212  005204                    INC     R4                              ; CHANNEL                                          9810
 22325 077214  020427  000043            CMP     R4,#43                          ; CHANNEL,*
 22326 077220  003002                    BGT     19$
 22327 077222  000167  177372            JMP     12$
 22328 077226  005205                19$:  INC     R5                              ; PLOG.SEL                                         9782
 22329 077230  020527  000006            CMP     R5,#6                           ; PLOG.SEL,*
 22330 077234  003002                    BGT     20$
 22331 077236  000167  177256            JMP     2$
 22332 077242  000207                20$:  RTS     PC                              ;                                                  9740
 22333
 22334                               ; Routine Size:  186 words
 22335                               ; Maximum stack depth per invocation:  18 words
 22340
 22341
 22345
 22346                                       .SBTTL  T53
 22350 077244                        T53::
 22351 077244  004767  177210        1$:     JSR     PC,$T53                         ;                                                  9846
 22352 077250  104466                        TRAP    66
 22353 077252  006000                        ROR     R0
 22354 077254  103773                        BLO     1$
 22355 077256  000207                        RTS     PC
 22356
 22357                               ;ML4                                                    14-May-1981 13:48:54    TOPS
 22358                               ;                                                       14-May-1981 13:05:08    PA:<
 22359
 22360                               ; Routine Size:  6 words
 22361                               ; Maximum stack depth per invocation:  0 words
 22366
 22367
 22368 ;          9849  !<BLF/PAGE>
```

```
22370 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
22371 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (104)
22372
22373 ;          9850  !
22374 ;          9851  BGNTST;
22375 ;          9852
22376 ;          9853  !++
22377 ;          9854  ! TEST NUMBER:  TST 54
22378 ;          9855  !
22379 ;          9856  ! TEST NAME:            SINGLE BIT ERROR SYNDRONE GENERATION & DECODE TEST
22380 ;          9857  !
22381 ;          9858  ! TEST DESCRIPTION:
22382 ;          9859  !       TEST SYNDRONE GENERATION, SYNDRONE DECODE
22383 ;          9860  !       AND ERROR CORRECTION FOR SINGLE BIT CHANNEL
22384 ;          9861  !       ERRORS BY:
22385 ;          9862  !
22386 ;          9863  !       1.  CLEAR THE FIRST CRC GROUP IN THE
22387 ;          9864  !           IO_BUF AND GOOD BLOCK TC ZEROES
22388 ;          9865  !
22389 ;          9866  !       2.  VIA DATA DIAGNOSTIC MODE FLIP A BIT
22390 ;          9867  !           TO A ONE
22391 ;          9868  !
22392 ;          9869  !       3.  THEN DO A MASS BUS READ TRANSFER AND
22393 ;          9870  !           EXAMIN THE IO BUF FOR ZEROES INDICATING
22394 ;          9871  !           THE CORRECTION WAS PERFORMED
22395 ;          9872  !
22396 ;          9873  !       4.  REPEAT WITH ALL BITS IN THE CRC GROUP
22397 ;          9874  !
22398 ;          9875  ! IMPLICIT INPUTS:
22399 ;          9876  !       IO_BUF
22400 ;          9877  !       A  VECTOR OF 256 WORDS WHERE DATA
22401 ;          9878  !       FOR MBUS READS AND WRITES TRANSFERS
22402 ;          9879  !       IS FOUND.
22403 ;          9880  !
22404 ;          9881  !       PD_TEMP
22405 ;          9882  !       A BIT VECTOR OF 16 BITS WHERE THE READ
22406 ;          9883  !       PROM DATA IS STORED AND ACCESSED FROM.
22407 ;          9884  !
22408 ;          9885  !
22409 ;          9886  !--
22410 ;          9887
22411 ;          9888  local
22412 ;          9889      DONE,                                 !DONE FLAG
22413 ;          9890      NIB_PAT : bitvector [4],              !STORAGE FOR SBE DATA GENERATED
22414 ;          9891      NIB_SEL,                              !POINTER WHERE SBE IS TO BE WRITTEN
22415 ;          9892      GD_WRD_CNT;                           !COUNT OF WHERE GOOD 'NIB_SEL' ARE FOUND
22416 ;          9893
22417 ;          9894  incr WRD_CNT from 0 to 5 do               !FCRCE SINGLE BIT ERRORS IN ONE CRC GROUP
22418 ;          9895      begin
22419 ;          9896
22420 ;          9897      incr BIT_CNT from 0 to 35 do          !FORCE SBE'S AT EACH BIT OF A WORD
22421 ;          9898          begin
22422 ;          9899
22423 ;          9900          incr CNT from 0 to 255 do         !CLEAR FIRST CRC GROUP OF IO_BUF
22424 ;          9901              IO_BUF [.CNT] = ZEROES;
22425 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
22426 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (104)
```

```
22427
22428 ;          9902
22429 ;          9903          CLR_MBUS;                                !CLEAR THE DRIVE
22430 ;          9904          GD_BLK_XFER ();                          !SET UP A GOOF BLOCK TRANSFER
22431 ;          9905          MLCS1 = write;                           !DO A WRITE TRANSFER
22432 ;          9906          TIME_OUT_LOOP;                           !WAIT UNTIL THE TRANSFER IS COMPLETE
22433 ;          9907          CLR_MBUS;                                !CLEAR THE DRIVE AGAIN
22434 ;          9908          DONE = ZERO;                             !CLEAR THE DONE FLAG
22435 ;          9909          GD_WRD_CNT = -1;                         !RESET THE COUNT
22436 ;          9910          MLD1 = ZEROES;                           !LOAD THE DATA DIAG REG (BITS 0-38)
22437 ;          9911          MLD2 = ZEROES;                           !WITH ZEROES PAT AND CRC BIT FOR
22438 ;          9912          MLE2 = %o'060000';                       !ZEROES PAT
22439 ;          9913          NIB_SEL = .BIT_CNT/4;                    !CALCULATE THE NIBBLE WHERE THE SBE IS TO GO
22440 ;          9914          DAT_DM_XFER ();                          !SET UP A DATA DIAG MODE TRANSFER
22441 ;          9915          MLCS1 = write;                           !DO A WRITE TRANSFER
22442 ;          9916          DELAY (ONE_US);                          !GIVE THE PROM DATA TIME TO GET OUT
22443 ;          9917
22444 ;          9918          do                                       !FIND THE GOOD NIB AT THE TESTED WORD
22445 ;          9919              begin
22446 ;          9920              PD_TEMP = .MLPD;                     !READ THE PROM DATA FOR THIS ARRAY WORD
22447 ;          9921
22448 ;          9922              if .PD_TEMP [.NIB_SEL] IS_SET        !IS THIS A GOOD NIBBLE
22449 ;          9923              then
22450 ;          9924                  DAT_CLK = ONE                    !IF NOT THEN GET THE NEXT ARRAY WORD
22451 ;          9925              else
22452 ;          9926                  begin                            !ELSE SEE IF WE ARE AT THE TESTED WORD
22453 ;          9927                  GD_WRD_CNT = .GD_WRD_CNT + 1;    !UP THE COUNT
22454 ;          9928                                                   !AND SEE IF WE ARE DONE
22455 ;          9929
22456 ;          9930                  if .GD_WRD_CNT eql .WRD_CNT then DONE = ONE else DAT_CLK = ONE;
22457 ;          9931
22458 ;          9932                  end;
22459 ;          9933
22460 ;          9934              end
22461 ;          9935          until .DONE IS_SET;                      !REPEAT UNTIL WE ARE DONE
22462 ;          9936
22463 ;          9937          NIB_PAT = ZEROES;                        !CLEAR THE SAVE LOCATION
22464 ;          9938          NIB_PAT [.BIT_CNT mod 4] = ONE;          !AND SET THE SBE IN THIS NIBBLE
22465 ;          9939          D1_TEMP = ZEROES;                        !CLEAR THE OTHER NIBBLES IN THIS ARRAY WORD
22466 ;          9940          D2_TEMP = ZEROES;                        !TO BE ZEROES PATTERN AND ZEROES CRC PATTERN
22467 ;          9941          E2_TEMP = %o'060000';
22468 ;          9942          LD_LNG_WRD (.NIB_SEL, .NIB_PAT);         !LOAD THE SBE INTO THE SAVE LOCATION
22469 ;          9943          WRT_LNG_WRD;                             !AND WRITE THE SBE INTO THE DATA DIAG RES'S
22470 ;          9944          DAT_CLK = ONE;                           !CLOCK THE SBE INTO THE ARRAYS MEMORY
22471 ;          9945          DELAY (ONE_US);                          !GIVE IT TIME TO WRITE INTO THE MEMORY
22472 ;          9946          BGNSUB;                                  !START OF SCOPE LOOP
22473 ;          9947          CLR_MBUS;                                !CLEAR OUT THE DATA DIAG MODE XFERR
22474 ;          9948          GD_BLK_XFER ();                          !SET UP A GOOD BLOCK TRANSFER
22475 ;          9949          ECC_DIS = ZERO;                          !CLEAR ECC DISABLE
22476 ;          9950          MLCS1 = read;                            !AND READ OUT THE SBE
22477 ;          9951          TIME_OUT_LOOP;
22478 ;          9952
22479 ;          9953          incr CNT from 0 to 13 do                 !SEE IF THE SBE WAS CORRECTED
22480 ;ML4                                                              14-May-1981 13:48:54    TOPS-20 Bliss-16 v2(212)
22481 ;                                                                 14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (104)
22482
22483 ;          9954
```

```
 22484 ;          9955              if .IO_BUF [.CNT] neq ZEROES        !TEST THE IO_BUF FOR ALL ZEROES
 22485 ;          9956              then
 22486 ;          9957                  begin                           !ERROR IF THE ERROR WAS NOT CORRECTED
 22487 ;          9958                  ERRDF (140, SYNC, DUMPER);
 22488 ;          9959                  PRINTB (THR_FMT, PHR_12, WRD_76, WRD_9);
 22489 ;          9960                  PRINTB (FMT_22, .WRD_CNT, .BIT_CNT);
 22490 ;          9961                  end;
 22491 ;          9962
 22492 ;          9963              ENDSUB;                             !END OF SCOPE LOOP
 22493 ;          9964              end;
 22494 ;          9965
 22495 ;          9966          end;
 22496 ;          9967
 22497 ;          9968      ENDTST;
 22501
 22502                                          .SBTTL  $T54
 22506 077260  004167  104632          $T54:   JSR     R1,$SAVE5                       ;                                       9848
 22507 077264  162706  000010                  SUB     #10,SP
 22508 077270  005001                          CLR     R1                              ; WRD.CNT                               9894
 22509 077272  005002                  1$:     CLR     R2                              ; BIT.CNT                               9897
 22510 077274  005003                  2$:     CLR     R3                              ; CNT                                   9900
 22511 077276  010304                  3$:     MOV     R3,R4                           ; CNT,*                                 9901
 22512 077300  006304                          ASL     R4
 22513 077302  005064  011600                  CLR     IO.BUF(R4)
 22514 077306  005203                          INC     R3                              ; CNT                                   9900
 22515 077310  020327  000377                  CMP     R3,#377                         ; CNT,*
 22516 077314  003770                          BLE     3$
 22517 077316  152777  000040  113670          BISB    #40,@ML.REG+40                  ;                                       9901
 22518 077324  016704  114252                  MOV     ML.DUT,R4
 22519 077330  042704  177770                  BIC     #177770,R4
 22520 077334  142777  000007  113652          BICB    #7,@ML.REG+40
 22521 077342  150477  113646                  BISB    R4,@ML.REG+40
 22522 077346  004767  116516                  JSR     PC,GD_BLK.XFER                  ;                                       9904
 22523 077352  012777  000061  113574          MOV     #61,@ML.REG                     ;                                       9905
 22524 077360  105777  113640          4$:     TSTB    @ML.REG+50
 22525 077364  100375                          BPL     4$
 22526 077366  152777  000040  113620          BISB    #40,@ML.REG+40                  ;                                       9906
 22527 077374  016704  114202                  MOV     ML.DUT,R4
 22528 077400  042704  177770                  BIC     #177770,R4
 22529 077404  142777  000007  113602          BICB    #7,@ML.REG+40
 22530 077412  150477  113576                  BISB    R4,@ML.REG+40
 22531 077416  005066  000002                  CLR     2(SP)                           ; DONE                                  9908
 22532 077422  012716  177777                  MOV     #-1,(SP)                        ; *,GD.WRD.CNT                          9909
 22533 077426  005077  113712                  CLR     @ML.REG+170                     ;                                       9910
 22534 077432  005077  113716                  CLR     @ML.REG+200                     ;                                       9911
 22535                                  ;ML4                                                    14-May-1981 13:48:54    TOPS
 22536                                  ;                                                       14-May-1981 13:05:08    PA:<
 22537
 22538 077436  012777  060000  113670          MOV     #60000,@ML.REG+160              ;                                       9912
 22539 077444  010246                          MOV     R2,-(SP)                        ; BIT.CNT,*                             9913
 22540 077446  012746  000004                  MOV     #4,-(SP)
 22541 077452  004767  104274                  JSR     PC,BL$DIV
 22542 077456  010005                          MOV     R0,R5                           ; *,NIB.SEL
 22543 077460  004767  114476                  JSR     PC,DAT.DM.XFER                  ;                                       9914
 22544 077464  012777  000061  113462          MOV     #61,@ML.REG                     ;                                       9915
 22545 077472  012703  000001                  MOV     #1,R3                           ; *,$$TMP2                              9916
 22546 077476  001411                  5$:     BEQ     8$
```

```
22547 077500  016704  102412                MOV     L$DLY,R4                ; *,$$TMP1
22548 077504  001404                        BEQ     7$
22549 077506  005066  000012          6$:   CLR     12(SP)                  ; $$TMP
22550 077512  005304                        DEC     R4                      ; $$TMP1
22551 077514  001374                        BNE     6$
22552 077516  005303                  7$:   DEC     R3                      ; $$TMP2
22553 077520  000766                        BR      5$
22554 077522  010504                  8$:   MOV     R5,R4                   ; NIB.SEL,*                     9922
22555 077524  006204                        ASR     R4
22556 077526  006204                        ASR     R4
22557 077530  006204                        ASR     R4
22558 077532  062704  013120                ADD     #PD.TEMP,R4
22559 077536  017767  113642  113354  9$:   MOV     @ML.REG+230,PD.TEMP     ;                               9920
22560 077544  010446                        MOV     R4,-(SP)                ;                               9922
22561 077546  010546                        MOV     R5,-(SP)                ; NIB.SEL,*
22562 077550  042716  177770                BIC     #177770,(SP)
22563 077554  012746  000001                MOV     #1,-(SP)
22564 077560  005046                        CLR     -(SP)
22565 077562  004767  103352                JSR     PC,BL$GT2
22566 077566  062706  000010                ADD     #10,SP
22567 077572  005300                        DEC     R0
22568 077574  001411                        BEQ     10$                     ;                               9924
22569 077576  005266  000004                INC     4(SP)                   ; GD.WRD.CNT                    9927
22570 077602  026601  000004                CMP     4(SP),R1                ; GD.WRD.CNT,WRD.CNT            9930
22571 077606  001004                        BNE     10$
22572 077610  012766  000001  000006        MOV     #1,6(SP)                ; *,DONE
22573 077616  000403                        BR      11$
22574 077620  152777  000020  113446  10$:  BISB    #20,@ML.REG+120
22575 077626  026627  000006  000001  11$:  CMP     6(SP),#1                ; DONE,*                        9935
22576 077634  001340                        BNE     9$
22577 077636  005066  000010                CLR     10(SP)                  ; NIB.PAT                       9937
22578 077642  010246                        MOV     R2,-(SP)                ; BIT.CNT,*                     9938
22579 077644  012746  000004                MOV     #4,-(SP)
22580 077650  004767  104110                JSR     PC,BL$MOD
22581 077654  010004                        MOV     R0,R4
22582 077656  006200                        ASR     R0
22583 077660  006200                        ASR     R0
22584 077662  006200                        ASR     R0
22585 077664  012703  000014                MOV     #14,R3
22586 077670  060603                        ADD     SP,R3                   ; NIB.PAT,*
22587 077672  060300                        ADD     R3,R0
22588 077674  010016                        MOV     R0,(SP)
22589 077676  010446                        MOV     R4,-(SP)
22590                               ;ML4                                                    14-May-1981 13:48:54    TOPS
22591                               ;                                                       14-May-1981 13:05:08    PA:<
22592
22593 077700  042716  177770                BIC     #177770,(SP)
22594 077704  012746  000001                MOV     #1,-(SP)
22595 077710  011646                        MOV     (SP),-(SP)
22596 077712  004767  103460                JSR     PC,BL$PU2
22597 077716  005067  111230                CLR     D1.TEMP                 ;                               9939
22598 077722  005067  111226                CLR     D2.TEMP                 ;                               9940
22599 077726  012767  060000  111222        MOV     #60000,E2.TEMP          ;                               9941
22600 077734  010516                        MOV     R5,(SP)                 ; NIB.SEL,*                     9942
22601 077736  016646  000022                MOV     22(SP),-(SP)            ; NIB.PAT,*
22602 077742  004767  117164                JSR     PC,LD.LNG.WRD
22603 077746  016777  111200  113370        MOV     D1.TEMP,@ML.REG+170
```

```
22604 077754  016777  111174  113372          MOV     D2.TEMP,@ML.REG+200
22605 077762  016777  111170  113344          MOV     E2.TEMP,@ML.REG+160
22606 077770  152777  000020  113276          BISB    #20,@ML.REG+120          ;                                        9944
22607 077776  012703  000001                  MOV     #1,R3                    ; *,$$TMP2                               9945
22608 100002  001411                  12$:    BEQ     15$
22609 100004  016704  102106                  MOV     L$DLY,R4                 ; *,$$TMP1
22610 100010  001404                  BEQ     14$
22611 100012  005066  000026          13$:    CLR     26(SP)                   ; $$TMP
22612 100016  005304                  DEC     R4                       ; $$TMP1
22613 100020  001374                  BNE     13$
22614 100022  005303                  14$:    DEC     R3                       ; $$TMP2
22615 100024  000766                  BR      12$
22616 100026  104402                  15$:    TRAP    2
22617 100030  152777  000040  113156          BISB    #40,@ML.REG+40           ;                                        9946
22618 100036  016704  113540                  MOV     ML.DUT,R4
22619 100042  042704  177770                  BIC     #177770,R4
22620 100046  142777  000007  113140          BICB    #7,@ML.REG+40
22621 100054  150477  113134                  BISB    R4,@ML.REG+40
22622 100060  004767  116004                  JSR     PC,GD.BLK.XFER           ;                                        9948
22623 100064  142777  000002  113202          BICB    #2,@ML.REG+120           ;                                        9949
22624 100072  012777  000071  113054          MOV     #71,@ML.REG              ;                                        9950
22625 100100  105777  113120          16$:    TSTB    @ML.REG+50
22626 100104  100375                  BPL     16$
22627 100106  005003                  CLR     R3                       ; CNT                                    9953
22628 100110  010304                  17$:    MOV     R3,R4                    ; CNT,*                                  9955
22629 100112  006304                  ASL     R4
22630 100114  005764  011600                  TST     IO.BUF(R4)
22631 100120  001432                  BEQ     18$
22632 100122  104455                  TRAP    55                       ;                                        9958
22633 100124  000214                  .WORD   214
22634 100126  010526                  .WORD   SYNC
22635 100130  024060                  .WORD   DUMPER
22636 100132  012746  006374          MOV     #WRD.9,-(SP)             ;                                        9959
22637 100136  012746  007340          MOV     #WRD.76,-(SP)
22638 100142  012746  007614          MOV     #PHR.12,-(SP)
22639 100146  012746  006034          MOV     #THR.FMT,-(SP)
22640 100152  012746  000004          MOV     #4,-(SP)
22641 100156  010600                  MOV     SP,R0                    ; SP,*
22642 100160  104414                  TRAP    14
22643 100162  010216                  MOV     R2,(SP)                  ; BIT.CNT,*                              9960
22644 100164  010146                  MOV     R1,-(SP)                 ; WRD.CNT,*
22645                         ;ML4                                                    14-May-1981 13:48:54    TOPS
22646                         ;                                                       14-May-1981 13:05:08    PA:<
22647
22648 100166  012746  005606          MOV     #FMT.22,-(SP)
22649 100172  012746  000003          MOV     #3,-(SP)
22650 100176  010600                  MOV     SP,R0                    ; SP,*
22651 100200  104414                  TRAP    14
22652 100202  062706  000020          ADD     #20,SP                   ;                                        9957
22653 100206  005203          18$:    INC     R3                       ; CNT                                    9953
22654 100210  020327  000015          CMP     R3,#15                   ; CNT,*
22655 100214  003735                  BLE     17$
22656 100216  104467                  TRAP    67                       ;                                        9961
22657 100220  006000                  ROR     R0
22658 100222  103701                  BLO     15$
22659 100224  062706  000020          ADD     #20,SP                   ;                                        9898
22660 100230  005202                  INC     R2                       ; BIT.CNT                                9897
```

```
 22661 100232  020227  000043                CMP     R2,#43                          ; BIT.CNT,*
 22662 100236  003002                        BGT     19$
 22663 100240  000167  177030                JMP     2$
 22664 100244  005201                19$:    INC     R1                              ; WRD.CNT                       9894
 22665 100246  020127  000005                CMP     R1,#5                           ; WRD.CNT,*
 22666 100252  003002                        BGT     20$
 22667 100254  000167  177012                JMP     1$
 22668 100260  062706  000010          20$:  ADD     #10,SP                          ;                               9848
 22669 100264  000207                        RTS     PC
 22670
 22671                                      ; Routine Size:  259 words
 22672                                      ; Maximum stack depth per invocation:  26 words
 22677
 22678
 22682
 22683                                              .SBTTL  T54
 22687 100266                               T54::
 22688 100266  004767  176766          1$:   JSR     PC,$T54                         ;                               9966
 22689 100272  104466                        TRAP    66
 22690 100274  006000                        ROR     R0
 22691 100276  103773                        BLO     1$
 22692 100300  000207                        RTS     PC
 22693
 22694                                      ; Routine Size:  6 words
 22695                                      ; Maximum stack depth per invocation:  0 words
 22700 ;ML4                                                  14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
 22701 ;                                                    14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (104)
 22702
 22703 ;          9969  !<BLF/PAGE>
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (105)

```
22705 ;ML4
22706 ;
22707
22708 ;          9970  !
22709 ;          9971  BGNTST;
22710 ;          9972
22711 ;          9973  !++
22712 ;          9974  ! TEST NUMBER:  TST 55
22713 ;          9975  !
22714 ;          9976  ! TEST NAME:            MULTIPLE BIT ERROR SYNDRONE GENERATION AND DECODE TEST
22715 ;          9977  !
22716 ;          9978  ! TEST DESCRIPTION:
22717 ;          9979  !       TEST SYNDRONE GENERATION, SYNDRONE DECODE
22718 ;          9980  !       AND ERROR CORRECTION FOR MULTIPLE BIT
22719 ;          9981  !       CHANNEL ERRORS BY:
22720 ;          9982  !
22721 ;          9983  !       1.  CLEAR THE FIRST CRC GROUP IN THE
22722 ;          9984  !           IO_BUF AND GOOD BLOCK TO ZEROES
22723 ;          9985  !
22724 ;          9986  !       2.  VIA DATA DIAG MODE FLIP ALL BITS IN
22725 ;          9987  !           A CHANNEL TO ONES
22726 ;          9988  !
22727 ;          9989  !       3.  THEN DO A MASS BUS READ TRANSFER AND
22728 ;          9990  !           EXAMIN THE IO BUF FOR ZEROES INDICATING
22729 ;          9991  !           THE CORRECTION WAS PERFORMED
22730 ;          9992  !
22731 ;          9993  !       4.  REPEAT FOR ALL CHANNELS IN THE CRC GROUP
22732 ;          9994  !
22733 ;          9995  ! IMPLICIT INPUTS:
22734 ;          9996  !       PD_TEMP
22735 ;          9997  !       A BIT VECTOR OF 16 BITS WHERE THE READ
22736 ;          9998  !       PROM DATA IS STORED OAND ACCESSED FROM.
22737 ;          9999  !
22738 ;          10000 !       IO_BUF
22739 ;          10001 !       A  VECTOR OF 256 WORDS WHERE DATA
22740 ;          10002 !       FOR MBUS READS AND WRITES TRANSFERS
22741 ;          10003 !       IS FOUND.
22742 ;          10004 !
22743 ;          10005 !
22744 ;          10006 !--
22745 ;          10007
22746 ;          10008 local
22747 ;          10009     NIB_PAT : bitvector [4],          !STORAGE FOR SBE TO BE GENERATED
22748 ;          10010     NIB_SEL,                          !POINTER TO WHERE ERROR IS TO BE WRITTEN
22749 ;          10011     GD_WRD_CNT;                       !COUNT OF WHERE GOOD 'NIB_SEL' ARE FOUND
22750 ;          10012
22751 ;          10013 E2_TEMP<12, 3> = %b'110';             !LOAD DATA STRUCTURE WITH ZEROES CRC PATTERN
22752 ;          10014
22753 ;          10015 incr BIT_CNT from 0 to 35 do          !TEST FOR MULTILPLE ERRORS AT 36 CHANNELS
22754 ;          10016     begin
22755 ;          10017     CLR_MBUS;                         !CLEAR THE DRIVE
22756 ;          10018
22757 ;          10019     incr CNT from 0 to 255 do         !CLEAR FIRST CRC GROUP IN THE IO_BUF
22758 ;          10020         IO_BUF [.CNT] = ZEROES;
22759 ;          10021
```

```
22761 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
22762 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (105)
22763
22764 ;          10022      GD_BLK_XFER ();                   !ALSO CLEAR THTE FIRST CRC GROUP
22765 ;          10023      MLCS1 = write;                    !IN THE ML11'S GOOD BLOCK
22766 ;          10024      TIME_OUT_LOOP;
22767 ;          10025      GD_WRD_CNT = ZERO;                !CLEAR COUNT
22768 ;          10026      D1_TEMP = ZEROES;                 !CLEAR D1_TEMP
22769 ;          10027      D2_TEMP = ZEROES;                 !CLEAR D2_TEMP
22770 ;          10028      NIB_PAT = ZEROES;                 !CLEAR NIBBLE PATTERN
22771 ;          10029      NIB_SEL = .BIT_CNT/4;             !CALCULATE THE NIBBLE TO BE IN ERROR
22772 ;          10030      NIB_PAT [.BIT_CNT mod 4] = ONE;   !CALCULATE AND THE BIT TO BE IN ERROR
22773 ;          10031      LD_LNG_WRD (.NIB_SEL, .NIB_PAT);  !LOAD THE ERROR IN D1_TEMP OR D2_TEMP
22774 ;          10032      WRT_LNG_WRD;                      !LOAD THE REGISTER WITH THE ERROR
22775 ;          10033      DAT_DM_XFER ();                   !SET UP A DATA DIAG MODE TRANSFER
22776 ;          10034      MLCS1 = write;                    !DO A WRITE TRANSFER
22777 ;          10035      DELAY (ONE_US);                   !GIVE THE PROM DATA TIME TO COME OUT
22778 ;          10036
22779 ;          10037      do                                !LOAD THIS CHANNEL WITH MULTIPLE ERRORS
22780 ;          10038          begin
22781 ;          10039          PD_TEMP = .MLPD;              !READ THIS ARRAY WORDS PROM_DATA
22782 ;          10040
22783 ;          10041          if .PD_TEMP [.NIB_SEL] IS_SET !SEE IF THIS A GOOD NIBBLE
22784 ;          10042          then
22785 ;          10043              DAT_CLK = ONE             !CLOCK OUT ANOTHER ARRAY WORD IF BAD
22786 ;          10044          else
22787 ;          10045              begin                     !ELSE CLOCK IN THE ERROR INTO A GOOD NIBBLE
22788 ;          10046              GD_WRD_CNT = .GD_WRD_CNT + 1;
22789 ;          10047              DAT_CLR = ONE;
22790 ;          10048              end;
22791 ;          10049
22792 ;          10050          end
22793 ;          10051      until .GD_WRD_CNT eql 6;          !REPEAT UNTIL 6 WORDS ARE WRITTEN
22794 ;          10052
22795 ;          10053      BGNSUB;                           !START OF SCOPE LOOP
22796 ;          10054      CLR_MBUS;                         !CLEAR OUT THE DATA DISG MODE TRANSFER
22797 ;          10055      GD_BLK_XFER ();                   !SET UP A GOOD BLOCK TRANSFER
22798 ;          10056      ECC_DIS = 0;                      !ENABLE ERROR CORRECTION
22799 ;          10057      MLCS1 = read;                     !READ OUT THE ERRORS
22800 ;          10058      TIME_OUT_LOOP;                    !WAIT FOR THE TRANSFER TO COMPLETE
22801 ;          10059
22802 ;          10060      incr WRD_CNT from 0 to 13 do      !SEE IF THE CORRECTIONS WHERE DONE
22803 ;          10061
22804 ;          10062          if .IO_BUF [.WRD_CNT] neq ZEROES  !TEST THE IO_BUF FOR ALL ZEROES
22805 ;          10063          then
22806 ;          10064              begin                     !REPORT THE ERROR IF ANY LOCATIONS ARE NOT ZEROES
22807 ;          10065              ERRDF (141, SYNC, DUMPER);
22808 ;          10066              PRINTB (THR_FMT, PHR_13, WRD_76, WRD_9);
22809 ;          10067              PRINTB (FMT_22, .WRD_CNT, .BIT_CNT);
22810 ;          10068              end;
22811 ;          10069
22812 ;          10070      ENDSUB;                           !END OF SCOPE LOOP
22813 ;          10071      end;
22814 ;          10072
22815 ;          10073 ENDTST;
```

```
22817 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
22818 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (105)
22819
22823
22824                                          .SBTTL  $T55
22828 100302  004167  103610          $T55:   JSR     R1,$SAVE5                ;                              9968
22829 100306  024646                  1$:     CMP     -(SP),-(SP)
22830 100310  042767  070000  110640          BIC     #70000,E2.TEMP           ;                              1001
22831 100316  052767  060000  110632          BIS     #60000,E2.TEMP
22832 100324  005003                          CLR     R3                       ; BIT.CNT                      1001
22833 100326  152777  000040  112660          BISB    #40,@ML.REG+40           ;                              1001
22834 100334  016702  113242                  MOV     ML.DUT,R2
22835 100340  042702  177770                  BIC     #177770,R2
22836 100344  142777  000007  112642          BICB    #7,@ML.REG+40
22837 100352  150277  112636                  BISB    R2,@ML.REG+40
22838 100356  005001                          CLR     R1                       ; CNT                          1001
22839 100360  010102                  2$:     MOV     R1,R2                    ; CNT,*                        1002
22840 100362  006302                          ASL     R2
22841 100364  005062  011600                  CLR     IO.BUF(R2)
22842 100370  005201                          INC     R1                       ; CNT                          1001
22843 100372  020127  000377                  CMP     R1,#377                  ; CNT,*
22844 100376  003770                          BLE     2$
22845 100400  004767  115464                  JSR     PC,GD.BLK.XFER           ;                              1002
22846 100404  012777  000061  112542          MOV     #61,@ML.REG              ;                              1002
22847 100412  105777  112606          3$:     TSTB    @ML.REG+50
22848 100416  100375                          BPL     3$
22849 100420  005005                          CLR     R5                       ; GD.WRD.CNT                   1002
22850 100422  005067  110524                  CLR     D1.TEMP                  ;                              1002
22851 100426  005067  110522                  CLR     D2.TEMP                  ;                              1002
22852 100432  005016                          CLR     (SP)                     ; NIB.PAT                      1002
22853 100434  010346                          MOV     R3,-(SP)                 ; BIT.CNT,*                    1002
22854 100436  012746  000004                  MOV     #4,-(SP)
22855 100442  004767  103304                  JSR     PC,BL$DIV
22856 100446  010004                          MOV     R0,R4                    ; *,NIB.SEL
22857 100450  010346                          MOV     R3,-(SP)                 ; BIT.CNT,*                    1003
22858 100452  012746  000004                  MOV     #4,-(SP)
22859 100456  004767  103302                  JSR     PC,BL$MOD
22860 100462  010002                          MOV     R0,R2
22861 100464  006200                          ASR     R0
22862 100466  006200                          ASR     R0
22863 100470  006200                          ASR     R0
22864 100472  012701  000010                  MOV     #10,R1
22865 100476  060601                          ADD     SP,R1                    ; NIB.PAT,*
22866 100500  060100                          ADD     R1,R0
22867 100502  010016                          MOV     R0,(SP)
22868 100504  010246                          MOV     R2,-(SP)
22869 100506  042716  177770                  BIC     #177770,(SP)
22870 100512  012746  000001                  MOV     #1,-(SP)
22871 100516  011646                          MOV     (SP),-(SP)
```

```
                                                                                                                14-May-1981 13:48:54    TOPS
                                                                                                                14-May-1981 13:05:08    PA:<
22873                                   ;ML4
22874                                   ;
22875
22876 100520  004767  102652                    JSR     PC,BL$PU2
22877 100524  010416                            MOV     R4,(SP)                 ; NIB.SEL,*                             1003
22878 100526  016646  000016                    MOV     16(SP),-(SP)            ; NIB.PAT,*
22879 100532  004767  116374                    JSR     PC,LD.LNG.WRD
22880 100536  016777  110410  112600            MOV     D1.TEMP,@ML.REG+170
22881 100544  016777  110404  112602            MOV     D2.TEMP,@ML.REG+200
22882 100552  016777  110400  112554            MOV     E2.TEMP,@ML.REG+160
22883 100560  004767  113376                    JSR     PC,DAT.DM.XFER          ;                                       1003
22884 100564  012777  000061  112362            MOV     #61,@ML.REG             ;                                       1003
22885 100572  012701  000001                    MOV     #1,R1                   ; *,$$TMP2                              1003
22886 100576  001411                    4$:     BEQ     7$
22887 100600  016702  101312                    MOV     L$DLY,R2                ; *,$$TMP1
22888 100604  001404                            BEQ     6$
22889 100606  005066  000022            5$:     CLR     22(SP)                  ; $$TMP
22890 100612  005302                            DEC     R2                      ; $$TMP1
22891 100614  001374                            BNE     5$
22892 100616  005301                    6$:     DEC     R1                      ; $$TMP2
22893 100620  000766                            BR      4$
22894 100622  010402                    7$:     MOV     R4,R2                   ; NIB.SEL,*                             1004
22895 100624  006202                            ASR     R2
22896 100626  006202                            ASR     R2
22897 100630  006202                            ASR     R2
22898 100632  062702  013120                    ADD     #PD.TEMP,R2
22899 100636  017767  112542  112254    8$:     MOV     @ML.REG+230,PD.TEMP     ;                                       1003
22900 100644  010246                            MOV     R2,-(SP)                ;                                       1004
22901 100646  010446                            MOV     R4,-(SP)                ; NIB.SEL,*
22902 100650  042716  177770                    BIC     #177770,(SP)
22903 100654  012746  000001                    MOV     #1,-(SP)
22904 100660  005046                            CLR     -(SP)
22905 100662  004767  102252                    JSR     PC,BL$GT2
22906 100666  062706  000010                    ADD     #10,SP
22907 100672  005300                            DEC     R0
22908 100674  001401                            BEQ     9$                      ;                                       1004
22909 100676  005205                            INC     R5                      ; GD.WRD.CNT                            1004
22910 100700  152777  000020  112366    9$:     BISB    #20,@ML.REG+120         ;                                       1004
22911 100706  020527  000006                    CMP     R5,#6                   ; GD.WRD.CNT,*                          1005
22912 100712  001351                            BNE     8$
22913 100714  104402                    10$:    TRAP    2
22914 100716  152777  000040  112270            BISB    #40,@ML.REG+40          ;                                       1005
22915 100724  016702  112652                    MOV     ML.DUT,R2
22916 100730  042702  177770                    BIC     #177770,R2
22917 100734  142777  000007  112252            BICB    #7,@ML.REG+40
22918 100742  150277  112246                    BISB    R2,@ML.REG+40
22919 100746  004767  115116                    JSR     PC,GD.BLK.XFER          ;                                       1005
22920 100752  142777  000002  112314            BICB    #2,@ML.REG+120          ;                                       1005
22921 100760  012777  000071  112166            MOV     #71,@ML.REG             ;                                       1005
22922 100766  105777  112232            11$:    TSTB    @ML.REG+50
22923 100772  100375                            BPL     11$
22924 100774  005001                            CLR     R1                      ; WRD.CNT                               1006
22925 100776  010102                    12$:    MOV     R1,R2                   ; WRD.CNT,*                             1006
22926 101000  006302                            ASL     R2
22927 101002  005762  011600                    TST     IO.BUF(R2)
```

```
                                                
```

```
22985                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
22986                                   ;                                                       14-May-1981 13:05:08    PA:<
22987
22988 101144  006000                              ROR     R0
22989 101146  103773                              BLO     1$
22990 101150  000207                              RTS     PC
22991
22992                                   ; Routine Size:  6 words
22993                                   ; Maximum stack depth per invocation:  0 words
22998
22999
23000 ;          10074 !<BLF/PAGE>
```

```
23002 ;ML4
23003 ;
23004
23005 ;          10075 !
23006 ;          10076 BGNTST;
23007 ;          10077
23008 ;          10078 !++
23009 ;          10079 ! TEST NUMBER:  TST 56
23010 ;          10080 !
23011 ;          10081 ! TEST NAME:            ECC ERROR REGISTER TEST
23012 ;          10082 !
23013 ;          10083 ! TEST DESCRIPTION:
23014 ;          10084 !       TEST THE ECC ERROR REGISTER FOR
23015 ;          10085 !       CLEARING AND LATCHING OF ECC ERROR
23016 ;          10086 !       INFORMATION ON DETTECTION
23017 ;          10087 !       OF ECC ERRORS BY:
23018 ;          10088 !
23019 ;          10089 !       1.  THIS TEST IS TABLE DRIVEN.  DATA TABLE (DT_1)
23020 ;          10090 !           CONTAINS ERROR FORCING DATA AND EXPECTED
23021 ;          10091 !           MLEE DATA AS A RESULT OF THE FORCED ERROR
23022 ;          10092 !
23023 ;          10093 !       2.  THE TEST THEREFOR INDEXES INTO DT_1 TO FORCE
23024 ;          10094 !           AN ECC ERROR
23025 ;          10095 !
23026 ;          10096 !       3.  A MASS BUS READ TRANSFER IS DONE
23027 ;          10097 !
23028 ;          10098 !       4.  DT_1 IS AGAIN INDEXED AND ITS CONTENTS
23029 ;          10099 !           IS COMPARED AGAINST THE RESULTING
23030 ;          10100 !           MLEE REGISTER  CONTENTS
23031 ;          10101 !
23032 ;          10102 !       5.  A MASS BUS CLEAR IS DONE AN MLEE IS CHECKED FOR
23033 ;          10103 !           ZEROES
23034 ;          10104 !
23035 ;          10105 !       6.  THE ECH BIT IS ALSO EXAMINED FOR CORRECT
23036 ;          10106 !           LATCHING INFORMATION
23037 ;          10107 !
23038 ;          10108 !
23039 ;          10109 !
23040 ;          10110 !--
23041 ;          10111
23042 ;          10112 local
23043 ;          10113     FINISH,                          !VARIABLE ENDING CONDITION FOR LATCH_LOOP
23044 ;          10114     ERR_FLG;                         !ERROR FLAG
23045 ;          10115
23046 ;          10116 CLR_MBUS;                            !CLEAR THE DRIVE
23047 ;          10117 FINISH = 4;                          !MAKE LATCH_LOOP DO 5 LOOPS AT FIRST
23048 ;          10118
23049 ;          10119 incr CLR_LOOP from 0 to 1 do         !THE FIRST PASS TESTS MLEE FOR LATCHING
23050 ;          10120     !THE SECOND PASS TESTS MLEE FOR CLEARING
23051 ;          10121     begin
23052 ;          10122
23053 ;          10123     incr LATCH_LOOP from 0 to .FINISH do   !TEST MLEE FOR LATCHING BY USING DT_1 DATA
23054 ;          10124         begin
23055 ;          10125         BGNSUB;                      !START OF SCOPE LOOP
23056 ;          10126         ERR_FLG = ZERO;              !CLEAR THE ERROR FLAG
23057 ;ML4
23058 ;
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (106)

```
23059
23060 ;          10127          MLER = ZEROES;                                  !CLEAR THE ERROR REGISTER
23061 ;          10128          ECC_DM = ONE;                                   !ENABLE ECC DIAG MODE
23062 ;          10129          PAR_CRC_WRD = .DT_1 [.LATCH_LOOP, P0_5];        !GET PAR_CRC_WRD DATA FROM DT_1
23063 ;          10130          CRC_A = .DT_1 [.LATCH_LOOP, A0_5];              !GET CRC_A FROM DT_1
23064 ;          10131          CRC_B = .DT_1 [.LATCH_LOOP, B0_5];              !GET CRC_B FROM DT_1
23065 ;          10132          GD_BLK_XFER ();                                 !SET UP A GOOD BLOCK TRANSFER
23066 ;          10133          ECC_DIS = ZERO;                                 !DISABLE ECC DIABLE
23067 ;          10134          MLCS1 = read;                                   !READ THE ERROR AND LACTH MLEE
23068 ;          10135          TIME_OUT_LOOP;
23069 ;          10136
23070 ;          10137          if .CRC_ERR neq .DT_1 [.LATCH_LOOP, CRC_DATA] then ERR_FLG = ONE;       !TEST CRC_ERR
23071 ;          10138
23072 ;          10139          if .SGL_ERR neq .DT_1 [.LATCH_LOOP, SGL_DATA] then ERR_FLG = ONE;       !TEST SGL_ERR
23073 ;          10140
23074 ;          10141          if .UNC_ERR neq .DT_1 [.LATCH_LOOP, UNC_DATA] then ERR_FLG = ONE;       !TEST UNC
23075 ;          10142
23076 ;          10143          if .BIT_IN_ERR neq .DT_1 [.LATCH_LOOP, E0_5] then ERR_FLG = ONE;        !TEST BIT IN ERROR
23077 ;          10144
23078 ;          10145          if .CHAN_IN_ERR neq .DT_1 [.LATCH_LOOP, C0_5] then ERR_FLG = ONE;       !TEST CHANNEL IN ERROR
23079 ;          10146
23080 ;          10147          if .ECH_ERR neq .DT_1 [.LATCH_LOOP, ECH_DATA]   !TEST IF ECH GOT SET
23081 ;          10148          then
23082 ;          10149              begin                                       !REPORT ERROR IF NOT SET
23083 ;          10150              ERRDF (123, ASYNC, DUMPER);
23084 ;          10151              PRINTB (FOR_FMT, WRD_67, WRD_10, WRD_69, WRD_14);
23085 ;          10152              end;
23086 ;          10153
23087 ;          10154          if .ERR_FLG IS_SET                              !TEST IF THE ERROR FLAG GOT SET
23088 ;          10155          then
23089 ;          10156              begin                                       !REPORT ERROR IF SET
23090 ;          10157              ERRDF (124, SYNC, DUMPER);
23091 ;          10158              PRINTB (THR_FMT, REG_14, WRD_52, PHR_11);
23092 ;          10159              PRINTB (FMT_1, .DT_1 [.LATCH_LOOP, EE_DATA], .MLEE);
23093 ;          10160              PRINTB (FMT_19, .DT_1 [.LATCH_LOOP, B0_5], .DT_1 [.LATCH_LOOP, A0_5], .DT_1 [.LATCH_LOOP, P0_5])
23094 ;          10161              end;
23095 ;          10162
23096 ;          10163              ENDSUB;                                     !END OF SCOPE LOOP
23097 ;          10164              end;
23098 ;          10165
23099 ;          10166          CLR_MBUS;                                       !CLEAR THE REGISTER
23100 ;          10167
23101 ;          10168          if .MLEE neq ZERO                               !SEE IF THE REGISTER GOT CLEARED
23102 ;          10169          then
23103 ;          10170              begin                                       !REPORT ERROR IF NOT CLEARED
23104 ;          10171              ERRDF (125, SYNC, DUMPER);
23105 ;          10172              PRINTB (FOR_FMT, REG_14, WRD_52, FNC_7, WRD_14);
23106 ;          10173              end;
23107 ;          10174
23108 ;          10175          ECC_DM = ONE;                                   !ENABLE ECC DISABLE
23109 ;          10176          FINISH = ZERO;                                  !MAKE LATCH_LOOP LOOP ONCE TO CLEAR MLEE AGAIN
23110 ;          10177          end;
23111 ;          10178
23112 ;ML4                                                                     14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
23113 ;                                                                        14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (106)
23114
23115 ;          10179 ENDTST;
```

```
23119
23120                                          .SBTTL  $T56
23124 101152  004167  102740          $T56:   JSR     R1,$SAVE5               ;                                1007
23125 101156  152777  000040  112030          BISB    #40,@ML.REG+40          ;                                1011
23126 101164  016705  112412                  MOV     ML.DUT,R5
23127 101170  042705  177770                  BIC     #177770,R5
23128 101174  142777  000007  112012          BICB    #7,@ML.REG+40
23129 101202  150577  112006                  BISB    R5,@ML.REG+40
23130 101206  012746  000004                  MOV     #4,-(SP)                ; *,FINISH                       1011
23131 101212  005046                          CLR     -(SP)                   ; CLR.LOOP                       1011
23132 101214  005004                  1$:     CLR     R4                      ; LATCH.LOOP                     1012
23133 101216  000167  000644                  JMP     16$
23134 101222  010446                  2$:     MOV     R4,-(SP)                ; LATCH.LOOP,*                   1012
23135 101224  012746  000006                  MOV     #6,-(SP)
23136 101230  004767  102272                  JSR     PC,BL$MUL
23137 101234  010005                          MOV     R0,R5
23138 101236  012701  013534                  MOV     #DT.1,R1                ;                                1013
23139 101242  060501                          ADD     R5,R1
23140 101244  012702  013536                  MOV     #DT.1+2,R2              ;                                1012
23141 101250  060502                          ADD     R5,R2
23142 101252  104402                  3$:     TRAP    2                       ;                                1012
23143 101254  005003                          CLR     R3                      ; ERR.FLG                        1012
23144 101256  005077  111752                  CLR     @ML.REG+60              ;                                1012
23145 101262  152777  000001  112004          BISB    #1,@ML.REG+120          ;                                1012
23146 101270  011200                          MOV     (R2),R0                 ;                                1012
23147 101272  006300                          ASL     R0
23148 101274  006300                          ASL     R0
23149 101276  042700  140377                  BIC     #140377,R0
23150 101302  042777  037400  112014          BIC     #37400,@ML.REG+150
23151 101310  050077  112010                  BIS     R0,@ML.REG+150
23152 101314  111200                          MOVB    (R2),R0                 ;                                1013
23153 101316  042700  177700                  BIC     #177700,R0
23154 101322  042777  000077  111774          BIC     #77,@ML.REG+150
23155 101330  050077  111770                  BIS     R0,@ML.REG+150
23156 101334  116500  013540                  MOVB    DT.1+4(R5),R0           ;                                1013
23157 101340  042700  177700                  BIC     #177700,R0
23158 101344  042777  000077  111762          BIC     #77,@ML.REG+160
23159 101352  050077  111756                  BIS     R0,@ML.REG+160
23160 101356  004767  114506                  JSR     PC,GD.BLK.XFER          ;                                1013
23161 101362  142777  000002  111704          BICB    #2,@ML.REG+120          ;                                1013
23162 101370  012777  000071  111556          MOV     #71,@ML.REG             ;                                1013
23163 101376  105777  111622          4$:     TSTB    @ML.REG+50
23164 101402  100375                          BPL     4$
23165 101404  011146                          MOV     (R1),-(SP)              ;                                1013
23166 101406  042716  157777                  BIC     #157777,(SP)
23167                                 ;ML4                                                    14-May-1981 13:48:54    TOPS
23168                                 ;                                                       14-May-1981 13:05:08    PA:<
23169
23170 101412  017700  111746                  MOV     @ML.REG+210,R0
23171 101416  042700  157777                  BIC     #157777,R0
23172 101422  020026                          CMP     R0,(SP)+
23173 101424  001402                          BEQ     5$
23174 101426  012703  000001                  MOV     #1,R3                   ; *,ERR.FLG
23175 101432  011146                  5$:     MOV     (R1),-(SP)              ;                                1013
23176 101434  042716  137777                  BIC     #137777,(SP)
23177 101440  017700  111720                  MOV     @ML.REG+210,R0
23178 101444  042700  137777                  BIC     #137777,R0
```

```
23179 101450 020026                    CMP     R0,(SP)+
23180 101452 001402                    BEQ     6$
23181 101454 012703 000001             MOV     #1,R3                   ; *,ERR.FLG
23182 101460 011146             6$:    MOV     (R1),-(SP)              ;                       1014
23183 101462 042716 077777             BIC     #77777,(SP)
23184 101466 017700 111672             MOV     @ML.REG+210,R0
23185 101472 042700 077777             BIC     #77777,R0
23186 101476 020026                    CMP     R0,(SP)+
23187 101500 001402                    BEQ     7$
23188 101502 012703 000001             MOV     #1,R3                   ; *,ERR.FLG
23189 101506 111146             7$:    MOVB    (R1),-(SP)              ;                       1014
23190 101510 042716 177700             BIC     #177700,(SP)
23191 101514 117700 111644             MOVB    @ML.REG+210,R0
23192 101520 042700 177700             BIC     #177700,R0
23193 101524 020026                    CMP     R0,(SP)+
23194 101526 001402                    BEQ     8$
23195 101530 012703 000001             MOV     #1,R3                   ; *,ERR.FLG
23196 101534 011146             8$:    MOV     (R1),-(SP)              ;                       1014
23197 101536 042716 170077             BIC     #170077,(SP)
23198 101542 017700 111616             MOV     @ML.REG+210,R0
23199 101546 042700 170077             BIC     #170077,R0
23200 101552 020026                    CMP     R0,(SP)+
23201 101554 001402                    BEQ     9$
23202 101556 012703 000001             MOV     #1,R3                   ; *,ERR.FLG
23203 101562 011246             9$:    MOV     (R2),-(SP)              ;                       1014
23204 101564 005046                    CLR     -(SP)
23205 101566 032766 010000 000002      BIT     #10000,2(SP)
23206 101574 001401                    BEQ     10$
23207 101576 005216                    INC     (SP)
23208 101600 005000             10$:   CLR     R0
23209 101602 132777 000100 111424      BITB    #100,@ML.REG+60
23210 101610 001401                    BEQ     11$
23211 101612 005200                    INC     R0
23212 101614 020026             11$:   CMP     R0,(SP)+
23213 101616 001002                    BNE     12$
23214 101620 005726                    TST     (SP)+
23215 101622 000425                    BR      13$
23216 101624 005726             12$:   TST     (SP)+
23217 101626 104455                    TRAP    55                      ;                       1015
23218 101630 000173                    .WORD   173
23219 101632 010464                    .WORD   ASYNC
23220 101634 024060                    .WORD   DUMPER
23221 101636 012746 006442             MOV     #WRD.14,-(SP)           ;                       1015
23222                           ;ML4                                                  14-May-1981 13:48:54    TOPS
23223                           ;                                                     14-May-1981 13:05:08    PA:<
23224
23225 101642 012746 007252             MOV     #WRD.69,-(SP)
23226 101646 012746 006406             MOV     #WRD.10,-(SP)
23227 101652 012746 007240             MOV     #WRD.67,-(SP)
23228 101656 012746 006046             MOV     #FOR.FMT,-(SP)
23229 101662 012746 000005             MOV     #5,-(SP)
23230 101666 010600                    MOV     SP,R0                   ; SP,*
23231 101670 104414                    TRAP    14
23232 101672 062706 000014             ADD     #14,SP                  ;                       1014
23233 101676 020327 000001      13$:   CMP     R3,#1                   ; ERR.FLG,*             1015
23234 101702 001062                    BNE     14$
23235 101704 104455                    TRAP    55                      ;                       1015
```

```
23236 101706  000174                    .WORD   174
23237 101710  010526                    .WORD   SYNC
23238 101712  024060                    .WORD   DUMPER
23239 101714  012746  007576            MOV     #PHR.11,-(SP)           ;                       1015
23240 101720  012746  007066            MOV     #WRD.52,-(SP)
23241 101724  012746  010376            MOV     #REG.14,-(SP)
23242 101730  012746  006034            MOV     #THR.FMT,-(SP)
23243 101734  012746  000004            MOV     #4,-(SP)
23244 101740  010600                    MOV     SP,R0                   ; SP,*
23245 101742  104414                    TRAP    14
23246 101744  017716  111414            MOV     @ML.REG+210,(SP)        ;                       1015
23247 101750  011146                    MOV     (R1),-(SP)
23248 101752  012746  004222            MOV     #FMT.1,-(SP)
23249 101756  012746  000003            MOV     #3,-(SP)
23250 101762  010600                    MOV     SP,R0                   ; SP,*
23251 101764  104414                    TRAP    14
23252 101766  011200                    MOV     (R2),R0                 ;                       1016
23253 101770  006200                    ASR     R0
23254 101772  006200                    ASR     R0
23255 101774  006200                    ASR     R0
23256 101776  006200                    ASR     R0
23257 102000  006200                    ASR     R0
23258 102002  006200                    ASR     R0
23259 102004  042700  177700            BIC     #177700,R0
23260 102010  010016                    MOV     R0,(SP)
23261 102012  111246                    MOVB    (R2),-(SP)
23262 102014  042716  177700            BIC     #177700,(SP)
23263 102020  116546  013540            MOVB    DT.1+4(R5),-(SP)
23264 102024  042716  177700            BIC     #177700,(SP)
23265 102030  012746  005412            MOV     #FMT.19,-(SP)
23266 102034  012746  000004            MOV     #4,-(SP)
23267 102040  010600                    MOV     SP,R0                   ; SP,*
23268 102042  104414                    TRAP    14
23269 102044  062706  000030            ADD     #30,SP                  ;                       1015
23270 102050  104467          14$:      TRAP    67                      ;                       1016
23271 102052  006000                    ROR     R0
23272 102054  103002                    BHIS    15$
23273 102056  000167  177170            JMP     3$
23274 102062  022626          15$:      CMP     (SP)+,(SP)+             ;                       1012
23275 102064  005204                    INC     R4                      ; LATCH.LOOP            1012
23276 102066  020466  000002    16$:    CMP     R4,2(SP)                ; LATCH.LOOP,FINISH
23277                         ;ML4                                                14-May-1981 13:48:54    TOPS
23278                         ;                                                   14-May-1981 13:05:08    PA:<
23279
23280 102072  003002                    BGT     17$
23281 102074  000167  177122            JMP     2$
23282 102100  152777  000040  111106  17$:    BISB    #40,@ML.REG+40          ;                       1016
23283 102106  016705  111470            MOV     ML.DUT,R5
23284 102112  042705  177770            BIC     #177770,R5
23285 102116  142777  000007  111070    BICB    #7,@ML.REG+40
23286 102124  150577  111064            BISB    R5,@ML.REG+40
23287 102130  005777  111230            TST     @ML.REG+210             ;                       1016
23288 102134  001424                    BEQ     18$
23289 102136  104455                    TRAP    55                      ;                       1017
23290 102140  000175                    .WORD   175
23291 102142  010526                    .WORD   SYNC
23292 102144  024060                    .WORD   DUMPER
```

```
23293 102146  012746  006442                MOV     #WRD.14,-(SP)                   ;                                       1017
23294 102152  012746  010012                MOV     #FNC.7,-(SP)
23295 102156  012746  007066                MOV     #WRD.52,-(SP)
23296 102162  012746  010376                MOV     #REG.14,-(SP)
23297 102166  012746  006046                MOV     #FOR.FMT,-(SP)
23298 102172  012746  000005                MOV     #5,-(SP)
23299 102176  010600                        MOV     SP,R0                           ; SP,*
23300 102200  104414                        TRAP    14
23301 102202  062706  000014                ADD     #14,SP                          ;                                       1017
23302 102206  152777  000001  111060  18$:  BISB    #1,@ML.REG+120                  ;                                       1017
23303 102214  005066  000002                CLR     2(SP)                           ; FINISH                                1017
23304 102220  005216                        INC     (SP)                            ; CLR.LOOP                              1011
23305 102222  021627  000001                CMP     (SP),#1                         ; CLR.LOOP,*
23306 102226  003002                        BGT     19$
23307 102230  000167  176760                JMP     1$
23308 102234  022626                19$:    CMP     (SP)+,(SP)+                     ;                                       1007
23309 102236  000207                        RTS     PC
23310
23311                               ; Routine Size:  283 words
23312                               ; Maximum stack depth per invocation:  22 words
23317
23318
23322
23323                                       .SBTTL  T56
23327 102240                        T56::
23328 102240  004767  176706        1$:     JSR     PC,$T56                         ;                                       1017
23329 102244  104466                        TRAP    66
23330 102246  006000                        ROR     R0
23331 102250  103773                        BLO     1$
23332                               ;ML4                                                    14-May-1981 13:48:54    TOPS
23333                               ;                                                       14-May-1981 13:05:08    PA:<
23334
23335 102252  000207                        RTS     PC
23336
23337                               ; Routine Size:  6 words
23338                               ; Maximum stack depth per invocation:  0 words
23343
23344
23345 ;          10180 !<BLF/PAGE>
```

```
23347 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
23348 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (107)
23349
23350 ;          10181 !
23351 ;          10182 BGNTST;
23352 ;          10183
23353 ;          10184 !++
23354 ;          10185 ! TEST NUMBER:  TST 57
23355 ;          10186 !
23356 ;          10187 ! TEST NAME:            ECC ERROR LOCATION REGISTER TEST
23357 ;          10188 !
23358 ;          10189 ! TEST DESCRIPTION:
23359 ;          10190 !       TEST THE ECC ERROR LOCATION REGISTER FOR CLOCKING, BIT UNIQUENESS
23360 ;          10191 !       CLEARING AND LATCHING OF DSA ADDRESSES ON DETECTION OF ECC ERROR BY:
23361 ;          10192 !
23362 ;          10193 !       1.  TEST FOR CLOCKING BY:
23363 ;          10194 !           A. DO A MASS BUS CLEAR FORCING REGISTER BITS TO ZEROES
23364 ;          10195 !           B. FORCE AN ECC ERROR AT THE LAST BLOCK AND DO A READ TRANSFER
23365 ;          10196 !           C. CHECK MLEL FOR LAST BLOCK ADDRESS
23366 ;          10197 !
23367 ;          10198 !       2.  TEST FOR BIT UNIQUENESS BY:
23368 ;          10199 !           A. WITH FORCED ECC ERRORS DO READ TRANSFERS AT VARIOUS DSA
23369 ;          10200 !              ADDRESSES AND CHECK MLEL FOR LATCHED DSA ADDRESSES
23370 ;          10201 !           B. DO CLEAR AND REPEAT
23371 ;          10202 !
23372 ;          10203 !       3.  TEST FOR LATCHING AND CLEARING BY:
23373 ;          10204 !           A. WITH A FORCED SINGLE BIT ERROR DO A READ TRANSFER AT THE
23374 ;          10205 !              LAST BLOCK. (LATCHES MLEL WITH THE LAST BLOCK ADDRESS TO START THE TEST OFF)
23375 ;          10206 !           B. AGAIN WITH A FORCED SINGLE BIT ERROR DO A READ TRANSFER AT DSA
23376 ;          10207 !              ZERO AND CHECK MLEL FOR NOT LATCHING.
23377 ;          10208 !              NO CLEAR DONE
23378 ;          10209 !           C. WITH FORCED UNCORRECTABLE ERRORS DO A READ TRANSFER AT DSA
23379 ;          10210 !              ZERO  CHECK MLEL FOR LATCHED DSA ZERO
23380 ;          10211 !              NO CLEAR DONE
23381 ;          10212 !           D. WITH FORCED SINGLE BIT ERRORS DO A READ TRANSFER AT THE LAST
23382 ;          10213 !              BLOCK AND CHECK MLEL FOR NOT LATCHING
23383 ;          10214 !              NO CLEAR DONE
23384 ;          10215 !           E. WITH A FORCED UNCORRECTABLE ERROR DO A READ TRANSFER AT
23385 ;          10216 !              THE LAST BLOCK AN CHECK MLEL FOR NOT LATCHING
23386 ;          10217 !           F. DO A MASS BUS CLEAR
23387 ;          10218 !           G. WITH A FORCED SINGLE BIT ERROR DO A READ TRANSFER AT THE
23388 ;          10219 !              LAST BLOCK
23389 ;          10220 !           H. DO A MASS BUS CLEAR AGAIN AND CHECK MLEL FOR ZEROES
23390 ;          10221 !
23391 ;          10222 ! IMPLICIT INPUTS:
23392 ;          10223 !       IO_BUF
23393 ;          10224 !       A VECTOR OF 256 WORDS WHERE DATA
23394 ;          10225 !       FOR MBUS READS AND WRITES TRANSFERS
23395 ;          10226 !       IS FOUND.
23396 ;          10227 !
23397 ;          10228 !
23398 ;          10229 !--
23399 ;          10230
23400 ;          10231 local
23401 ;          10232     SGL.                                    !SINGLE BIT ERROR PARRAMETER
```

```
23403 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
23404 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (107)
23405
23406 ;          10233      UNC,                              !UNCORRECTABLE ERR PARRAMETER
23407 ;          10234      SAVE,                             !TEMPORY SAVE LOCATION
23408 ;          10235      SHF_DATA,                         !SHIFTING DATA PATTERN SAVE LOCATION
23409 ;          10236      DSA_PAT;                          !EXPECTED DSA PATTERN LATCHED IN MLEL
23410 ;          10237
23411 ;          10238 BGNSUB;                                !START OF SCOPE LOOP
23412 ;          10239 CLR_MBUS;                              !CLEAR THE MLEL REGISTER
23413 ;          10240 FOR_ECC_ERR (SGL = 1, UNC = 0);        !FORCE A SBE
23414 ;          10241 MLDA = .LST_BLK;                       !LATCH THE LAST BLK ADRS INTO THE MLEL
23415 ;          10242 MLWC = not 255;
23416 ;          10243 MLBA = IO_BUF;
23417 ;          10244 MLCS1 = read;                          !DO A READ AND LATCH THE MLEL
23418 ;          10245 TIME_OUT_LOOP;
23419 ;          10246
23420 ;          10247 if .MLEL eql ZEROES                    !SEE IF THE MLEL GOT CLOCKED
23421 ;          10248 then
23422 ;          10249     begin                              !REPORT AN ERROR IF NOT CLOCKED
23423 ;          10250     ERRDF (126, SYNC, DUMPER);
23424 ;          10251     PRINTB (THR_FMT, REG_15, WRD_52, PHR_11);
23425 ;          10252     EXIT_TST;
23426 ;          10253     end;
23427 ;          10254
23428 ;          10255 ENDSUB;                                !END OF SCOPE LOOP
23429 ;          10256 SHF_DATA = .LST_ARR;                   !LOAD  THE SHIFT DATA
23430 ;          10257
23431 ;          10258 incr LOOP from 0 to 16 do              !FIRST TEST FOR 1'S AND 0'S THEN SHIFT DATA
23432 ;          10259     begin
23433 ;          10260     BGNSUB;                            !START OF SCOPE LOOP
23434 ;          10261     CLR_MBUS;                          !CLEAR THE DRIVE
23435 ;          10262     FOR_ECC_ERR (SGL = 1, UNC = 0);    !FORCE A SBE
23436 ;          10263
23437 ;          10264     selectone .LOOP of                 !SELECT THE TEST PATTERN
23438 ;          10265         set
23439 ;          10266
23440 ;          10267         [0] :                          !ZEROES PATTERN
23441 ;          10268             begin
23442 ;          10269             DSA_PAT = ZEROES;
23443 ;          10270             MLDA = .DSA_PAT;
23444 ;          10271             end;
23445 ;          10272
23446 ;          10273         [1] :                          !MOSTLY ONES PATTERN
23447 ;          10274             begin
23448 ;          10275             DSA_PAT = .LST_BLK;
23449 ;          10276             MLDA = .DSA_PAT;
23450 ;          10277             end;
23451 ;          10278
23452 ;          10279         [otherwise] :                  !SHIFT PATTERN
23453 ;          10280             begin
23454 ;          10281             DSA_PAT = .SHF_DATA;
23455 ;          10282             MLDA = .SHF_DATA;
23456 ;          10283             SHF_DATA = .SHF_DATA^-ONE;
23457 ;          10284             end;
```

```
23450 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
23460 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (107)
23461
23462 ;          10285          tes;
23463 ;          10286
23464 ;          10287      MLWC = not 255;                   !SET UP THE TRANSFER
23465 ;          10288      MLBA = IO_BUF;
23466 ;          10289      MLCS1 = read;                     !DO THE TRANSFER AND LATCH MLEL
23467 ;          10290      TIME_OUT_LOOP;
23468 ;          10291      SAVE = .MLEL;                     !READ AND SAVE THE MLEL REGISTER
23469 ;          10292
23470 ;          10293      if .SAVE neq .DSA_PAT             !SEE IF SAVE IS WHAT WE EXPECTED
23471 ;          10294      then
23472 ;          10295          begin                         !REPORT AN ERROR IF NOT THE SAME
23473 ;          10296          ERRDF (127, ASYNC, DUMPER);
23474 ;          10297          PRINTB (TWO_FMT, REG_15, PHR_4);
23475 ;          10298          PRINTB (FMT_2, .DSA_PAT, .SAVE, (.DSA_PAT xor .SAVE));
23476 ;          10299          end;
23477 ;          10300
23478 ;          10301      ENDSUB;                           !END OF SCOPE LOOP
23479 ;          10302      end;
23480 ;          10303
23481 ;          10304 CLR_MBUS;                              !CLEAR THE DRIVE
23482 ;          10305 FOR_ECC_ERR (SGL = 1, UNC = 0);        !FORCE A SBE
23483 ;          10306 MLWC = not 255;                        !SET UP A LAST BLOCK TRANSFER
23484 ;          10307 MLBA = IO_BUF;                         !TO LATCH THE MLEL WITH T
23485 ;          10308 MLDA = .LST_BLK;
23486 ;          10309 MLCS1 = read;
23487 ;          10310 TIME_OUT_LOOP;
23488 ;          10311
23489 ;          10312 incr LOOP from 0 to 4 do               !NOW TEST FOR CORRECT LATCHING
23490 ;          10313      begin
23491 ;          10314      BGNSUB;                           !START OF SCOPE LOOP
23492 ;          10315      MLER = ZEROES;                    !CLEAR THE ERROR REGISTER
23493 ;          10316
23494 ;          10317      case .LOOP from 0 to 4 of         !SELECT TEST DATA
23495 ;          10318          set
23496 ;          10319
23497 ;          10320          [0] :                         !FORCE A SBE AND CHECK FOR THE LST BLK ADRS
23498 ;          10321              begin
23499 ;          10322              SGL = 1;
23500 ;          10323              UNC = 0;
23501 ;          10324              DSA_PAT = .LST_BLK;
23502 ;          10325              MLDA = ZEROES;
23503 ;          10326              end;
23504 ;          10327
23505 ;          10328          [1] :                         !FORCE A UNC ERROR AND CHECK FOR ADRS ZERO
23506 ;          10329              begin
23507 ;          10330              SGL = 0;
23508 ;          10331              UNC = 1;
23509 ;          10332              DSA_PAT = ZEROES;
23510 ;          10333              MLDA = ZEROES;
23511 ;          10334              end;
23512 ;          10335
23513 ;          10336          [2] :                         !FORCE SBE AND CHECK FOR ADRS ZERO
```

```
23515 ;ML4
23516 ;
23517 ;
23518 ;          10337              begin
23519 ;          10338              SGL = 1;
23520 ;          10339              UNC = 0;
23521 ;          10340              DSA_PAT = ZEROES;
23522 ;          10341              MLDA = .LST_BLK;
23523 ;          10342              end;
23524 ;          10343
23525 ;          10344          [3] :                                  !FORCE UNC ERROR AND CHECK FOR ADRS ZERO
23526 ;          10345              begin
23527 ;          10346              SGL = 0;
23528 ;          10347              UNC = 1;
23529 ;          10348              DSA_PAT = ZEROES;
23530 ;          10349              MLDA = .LST_BLK;
23531 ;          10350              end;
23532 ;          10351
23533 ;          10352          [4] :                                  !FORCE SBE AND CHECK FOR ADRS ZERO
23534 ;          10353              begin
23535 ;          10354              CLR_MBUS;
23536 ;          10355              SGL = 1;
23537 ;          10356              UNC = 0;
23538 ;          10357              DSA_PAT = ZEROES;
23539 ;          10358              MLDA = .LST_BLK;
23540 ;          10359              end;
23541 ;          10360          tes;
23542 ;          10361
23543 ;          10362      FOR_ECC_ERR (.SGL, .UNC);                  !FORCE SELECTED FORCED ERROR INTO ML11
23544 ;          10363      MLWC = not 255;                            !SET UP A TRANSFER
23545 ;          10364      MLBA = IO_BUF;
23546 ;          10365      MLCS1 = read;                              !READ THE ERROR OUT AND LATCH MLEL
23547 ;          10366      TIME_OUT_LOOP;
23548 ;          10367
23549 ;          10368      if .LOOP neq 4                             !TEST THE REG FOR LATCH DATA IF LOOP NEQ 4
23550 ;          10369      then
23551 ;          10370          SAVE = .MLEL
23552 ;          10371      else
23553 ;          10372          begin                                  !ELSE TEST THE REG FOR CLEAR
23554 ;          10373          CLR_MBUS;
23555 ;          10374          SAVE = .MLEL;
23556 ;          10375          end;
23557 ;          10376
23558 ;          10377      if .SAVE neq .DSA_PAT                      !SEE IF THE REG LATCHED  OK
23559 ;          10378      then
23560 ;          10379          begin                                  !REPORT ERROR IF NOT LATCHED
23561 ;          10380          ERRDF (128, SYNC, DUMPER);
23562 ;          10381          PRINTB (THR_FMT, REG_15, WRD_71, WRD_10);
23563 ;          10382          PRINTB (FMT_1, .DSA_PAT, .SAVE)
23564 ;          10383          end;
23565 ;          10384
23566 ;          10385      ENDSUB;                                    !END OF SCOPE LOOP
23567 ;          10386      end;
23568 ;          10387
23569 ;          10388 ENDTST;
```

```
23571 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
23572 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (107)
23573
23577
23578                                          .SBTTL  $T57
23582 102254  004167  101636          $T57:   JSR     R1,$SAVE5                       ;                          1017
23583 102260  005746                          TST     -(SP)
23584 102262  104402                  1$:     TRAP    2                               ;                          1023
23585 102264  152777  000040  110722          BISB    #40,@ML.REG+40                  ;                          1023
23586 102272  016705  111304                  MOV     ML.DUT,R5
23587 102276  042705  177770                  BIC     #177770,R5
23588 102302  142777  000007  110704          BICB    #7,@ML.REG+40
23589 102310  150577  110700                  BISB    R5,@ML.REG+40
23590 102314  012703  000001                  MOV     #1,R3                           ; *,SGL                    1024
23591 102320  010346                          MOV     R3,-(SP)                        ; SGL,*
23592 102322  005001                          CLR     R1                              ; UNC
23593 102324  005046                          CLR     -(SP)
23594 102326  004767  113426                  JSR     PC,FOR.ECC.ERR
23595 102332  016777  107234  110644          MOV     LST.BLK,@ML.REG+30              ;                          1024
23596 102340  012777  177400  110616          MOV     #-400,@ML.REG+10                ;                          1024
23597 102346  012777  011600  110620          MOV     #IO.BUF,@ML.REG+20              ;                          1024
23598 102354  012777  000071  110572          MOV     #71,@ML.REG                     ;                          1024
23599 102362  105777  110636          2$:     TSTB    @ML.REG+50
23600 102366  100375                          BPL     2$
23601 102370  005777  111000                  TST     @ML.REG+220                     ;                          1024
23602 102374  001025                          BNE     3$
23603 102376  104455                          TRAP    55                              ;                          1025
23604 102400  000176                          .WORD   176
23605 102402  010526                          .WORD   SYNC
23606 102404  024060                          .WORD   DUMPER
23607 102406  012746  007576                  MOV     #PHR.11,-(SP)                   ;                          1025
23608 102412  012746  007066                  MOV     #WRD.52,-(SP)
23609 102416  012746  010404                  MOV     #REG.15,-(SP)
23610 102422  012746  006034                  MOV     #THR.FMT,-(SP)
23611 102426  012746  000004                  MOV     #4,-(SP)
23612 102432  010600                          MOV     SP,R0                           ; SP,*
23613 102434  104414                          TRAP    14
23614 102436  104463                          TRAP    63
23615 102440  062706  000016                  ADD     #16,SP                          ;                          1024
23616 102444  000167  001006                  JMP     26$                             ;                          1024
23617 102450  022626                  3$:     CMP     (SP)+,(SP)+                     ;                          1023
23618 102452  104467                          TRAP    67                              ;                          1025
23619 102454  006000                          ROR     R0
23620 102456  103701                          BLO     1$
23621 102460  016705  107112                  MOV     LST.ARR,R5                      ; *,SHF.DATA               1025
23622 102464  005002                          CLR     R2                              ; LOOP                     1025
23623 102466  104402                  4$:     TRAP    2                               ;                          1025
23624 102470  152777  000040  110516          BISB    #40,@ML.REG+40                  ;                          1026
23625 102476  016700  111100                  MOV     ML.DUT,R0
```

```
23627                                   ,ML4                                                    14-May-1981 13:48:54    TOPS
23628                                   :                                                       14-May-1981 13:05:08    PA:<
23629
23630 102502  042700  177770                    BIC     #177770,R0
23631 102506  142777  000007  110500            BICB    #7,@ML.REG+40
23632 102514  150077  110474                    BISB    R0,@ML.REG+40
23633 102520  012703  000001                    MOV     #1,R3                   ; *,SGL                 1026
23634 102524  010346                            MOV     R3,-(SP)                ; SGL,*
23635 102526  005001                            CLR     R1                      ; UNC
23636 102530  005046                            CLR     -(SP)
23637 102532  004767  113222                    JSR     PC,FOR.ECC.ERR
23638 102536  005702                            TST     R2                      ; LOOP                  1026
23639 102540  001002                            BNE     5$
23640 102542  005004                            CLR     R4                      ; DSA.PAT               1026
23641 102544  000405                            BR      6$                      ;                       1027
23642 102546  020227  000001          5$:       CMP     R2,#1                   ; LOOP,*                1026
23643 102552  001005                            BNE     7$
23644 102554  016704  107012                    MOV     LST.BLK,R4              ; *,DSA.PAT             1027
23645 102560  010477  110420          6$:       MOV     R4,@ML.REG+30           ; DSA.PAT,*             1027
23646 102564  000404                            BR      8$                      ;                       1026
23647 102566  010504                  7$:       MOV     R5,R4                   ; SHF.DATA,DSA.PAT      1028
23648 102570  010577  110410                    MOV     R5,@ML.REG+30           ; SHF.DATA,*            1028
23649 102574  006205                            ASR     R5                      ; SHF.DATA              1028
23650 102576  012777  177400  110360  8$:       MOV     #-400,@ML.REG+10        ;                       1028
23651 102604  012777  011600  110362            MOV     #IO.BUF,@ML.REG+20      ;                       1028
23652 102612  012777  000071  110334            MOV     #71,@ML.REG             ;                       1028
23653 102620  105777  110400          9$:       TSTB    @ML.REG+50
23654 102624  100375                            BPL     9$
23655 102626  017766  110542  000004            MOV     @ML.REG+220,4(SP)       ; *,SAVE                1029
23656 102634  026604  000004                    CMP     4(SP),R4                ; SAVE,DSA.PAT          1029
23657 102640  001441                            BEQ     10$
23658 102642  104455                            TRAP    55                      ;                       1029
23659 102644  000177                            .WORD   177
23660 102646  010464                            .WORD   ASYNC
23661 102650  024060                            .WORD   DUMPER
23662 102652  012746  007454                    MOV     #PHR.4,-(SP)            ;                       1029
23663 102656  012746  010404                    MOV     #REG.15,-(SP)
23664 102662  012746  006024                    MOV     #TWO.FMT,-(SP)
23665 102666  012746  000003                    MOV     #3,-(SP)
23666 102672  010600                            MOV     SP,R0                   ; SP,*
23667 102674  104414                            TRAP    14
23668 102676  016616  000014                    MOV     14(SP),(SP)             ; SAVE,*                1029
23669 102702  010446                            MOV     R4,-(SP)                ; DSA.PAT,*
23670 102704  046616  000002                    BIC     2(SP),(SP)
23671 102710  040466  000002                    BIC     R4,2(SP)                ; DSA.PAT,*
23672 102714  052616                            BIS     (SP)+,(SP)
23673 102716  016646  000014                    MOV     14(SP),-(SP)            ; SAVE,*
23674 102722  010446                            MOV     R4,-(SP)                ; DSA.PAT,*
23675 102724  012746  004266                    MOV     #FMT.2,-(SP)
23676 102730  012746  000004                    MOV     #4,-(SP)
23677 102734  010600                            MOV     SP,R0                   ; SP,*
23678 102736  104414                            TRAP    14
23679 102740  062706  000020                    ADD     #20,SP                  ;                       1029
23680 102744  022626                  10$:      CMP     (SP)+,(SP)+             ;                       1025
23681 102746  104467                            TRAP    67                      ;                       1029
```

```
23683                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
23684                                   ;                                                       14-May-1981 13:05:08    PA:<
23685
23686 102750  006000                            ROR     R0
23687 102752  103645                            BLO     4$
23688 102754  005202                            INC     R2                      ; LOOP                                  1025
23689 102756  020227  000020                    CMP     R2,#20                  ; LOOP,*
23690 102762  003641                            BLE     4$
23691 102764  152777  000040  110222            BISB    #40,@ML.REG+40          ;                                       1030
23692 102772  016705  110604                    MOV     ML.DUT,R5
23693 102776  042705  177770                    BIC     #177770,R5
23694 103002  142777  000007  110204            BICB    #7,@ML.REG+40
23695 103010  150577  110200                    BISB    R5,@ML.REG+40
23696 103014  012703  000001                    MOV     #1,R3                   ; *,SGL                                 1030
23697 103020  010346                            MOV     R3,-(SP)                ; SGL,*
23698 103022  005001                            CLR     R1                      ; UNC
23699 103024  005046                            CLR     -(SP)
23700 103026  004767  112726                    JSR     PC,FOR.ECC.ERR
23701 103032  012777  177400  110124            MOV     #-400,@ML.REG+10        ;                                       1030
23702 103040  012777  011600  110126            MOV     #IO.BUF,@ML.REG+20      ;                                       1030
23703 103046  016777  106520  110130            MOV     LST.BLK,@ML.REG+30      ;                                       1030
23704 103054  012777  000071  110072            MOV     #71,@ML.REG             ;                                       1030
23705 103062  105777  110136          11$:      TSTB    @ML.REG+50
23706 103066  100375                            BPL     11$
23707 103070  005002                            CLR     R2                      ; LOOP                                  1031
23708 103072  104402                  12$:      TRAP    2                       ;                                       1031
23709 103074  005077  110134                    CLR     @ML.REG+60              ;                                       1031
23710 103100  010205                            MOV     R2,R5                   ; LOOP,*                                1031
23711 103102  006305                            ASL     R5
23712 103104  066507  103110                    ADD     13$(R5),PC
23713 103110  000012                  13$:      .WORD   14$-13$
23714 103112  000026                            .WORD   15$-13$
23715 103114  000044                            .WORD   17$-13$
23716 103116  000046                            .WORD   18$-13$
23717 103120  000056                            .WORD   19$-13$
23718 103122  012703  000001          14$:      MOV     #1,R3                   ; *,SGL                                 1032
23719 103126  005001                            CLR     R1                      ; UNC                                   1032
23720 103130  016704  106436                    MOV     LST.BLK,R4              ; *,DSA.PAT                             1032
23721 103134  000404                            BR      16$                     ;                                       1032
23722 103136  005003                  15$:      CLR     R3                      ; SGL                                   1033
23723 103140  012701  000001                    MOV     #1,R1                   ; *,UNC                                 1033
23724 103144  005004                            CLR     R4                      ; DSA.PAT                               1033
23725 103146  005077  110032          16$:      CLR     @ML.REG+30              ;                                       1033
23726 103152  000430                            BR      22$                     ;                                       1031
23727 103154  000420                  17$:      BR      20$                     ;                                       1033
23728 103156  005003                  18$:      CLR     R3                      ; SGL                                   1034
23729 103160  012701  000001                    MOV     #1,R1                   ; *,UNC                                 1034
23730 103164  000417                            BR      21$                     ;                                       1034
23731 103166  152777  000040  110020  19$:      BISB    #40,@ML.REG+40          ;                                       1035
23732 103174  016705  110402                    MOV     ML.DUT,R5
23733 103200  042705  177770                    BIC     #177770,R5
23734 103204  142777  000007  110002            BICB    #7,@ML.REG+40
23735 103212  150577  107776                    BISB    R5,@ML.REG+40
23736 103216  012703  000001          20$:      MOV     #1,R3                   ; *,SGL                                 1035
23737 103222  005001                            CLR     R1                      ; UNC                                   1035
```

```
23739                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
23740                                   ;                                                       14-May-1981 13:05:08    PA:<
23741
23742 103224  005004                    21$:    CLR     R4                      ; DSA.PAT                               1035
23743 103226  016777  106340  107750            MOV     LST.BLK,@ML.REG+30      ;                                       1035
23744 103234  010346                    22$:    MOV     R3,-(SP)                ; SGL,*                                 1036
23745 103236  010146                            MOV     R1,-(SP)                ; UNC,*
23746 103240  004767  112514                    JSR     PC,FOR.ECC.ERR
23747 103244  012777  177400  107712            MOV     #-400,@ML.REG+10        ;                                       1036
23748 103252  012777  011600  107714            MOV     #IO.BUF,@ML.REG+20      ;                                       1036
23749 103260  012777  000071  107666            MOV     #71,@ML.REG             ;                                       1036
23750 103266  105777  107732            23$:    TSTB    @ML.REG+50
23751 103272  100375                            BPL     23$
23752 103274  020227  000004                    CMP     R2,#4                   ; LOOP,*                                1036
23753 103300  001014                            BNE     24$                     ;                                       1037
23754 103302  152777  000040  107704            BISB    #40,@ML.REG+40          ;                                       1037
23755 103310  016705  110266                    MOV     ML.DUT,R5
23756 103314  042705  177770                    BIC     #177770,R5
23757 103320  142777  000007  107666            BICB    #7,@ML.REG+40
23758 103326  150577  107662                    BISB    R5,@ML.REG+40
23759 103332  017766  110036  000010  24$:      MOV     @ML.REG+220,10(SP)      ; *,SAVE                                1037
23760 103340  026604  000010                    CMP     10(SP),R4               ; SAVE,DSA.PAT                          1037
23761 103344  001433                            BEQ     25$
23762 103346  104455                            TRAP    55                      ;                                       1038
23763 103350  000200                            .WORD   200
23764 103352  010526                            .WORD   SYNC
23765 103354  024060                            .WORD   DUMPER
23766 103356  012746  006406                    MOV     #WRD.10,-(SP)           ;                                       1038
23767 103362  012746  007272                    MOV     #WRD.71,-(SP)
23768 103366  012746  010404                    MOV     #REG.15,-(SP)
23769 103372  012746  006034                    MOV     #THR.FMT,-(SP)
23770 103376  012746  000004                    MOV     #4,-(SP)
23771 103402  010600                            MOV     SP,R0                   ; SP,*
23772 103404  104414                            TRAP    14
23773 103406  016616  000022                    MOV     22(SP),(SP)             ; SAVE,*                                1038
23774 103412  010446                            MOV     R4,-(SP)                ; DSA.PAT,*
23775 103414  012746  004222                    MOV     #FMT.1,-(SP)
23776 103420  012746  000003                    MOV     #3,-(SP)
23777 103424  010600                            MOV     SP,R0                   ; SP,*
23778 103426  104414                            TRAP    14
23779 103430  062706  000020                    ADD     #20,SP                  ;                                       1037
23780 103434  022626                    25$:    CMP     (SP)+,(SP)+             ;                                       1031
23781 103436  104467                            TRAP    67                      ;                                       1038
23782 103440  006000                            ROR     R0
23783 103442  103613                            BLO     12$
23784 103444  005202                            INC     R2                      ; LOOP                                  1031
23785 103446  020227  000004                    CMP     R2,#4                   ; LOOP,*
23786 103452  003607                            BLE     12$
23787 103454  022626                            CMP     (SP)+,(SP)+             ;                                       1017
23788 103456  005726                    26$:    TST     (SP)+
23789 103460  000207                            RTS     PC
23790
23791                                   ; Routine Size:  323 words
23792                                   ; Maximum stack depth per invocation:  19 words
```

```
23801
23802
23806
23807                                   .SBTTL  T57
23811 103462                    T57::
23812 103462  004767  176566  1$:       JSR     PC,$T57                         ;                1038
23813 103466  104466                    TRAP    66
23814 103470  006000                    ROR     R0
23815 103472  103773                    BLO     1$
23816 103474  000207                    RTS     PC
23817
23818                           ; Routine Size:  6 words
23819                           ; Maximum stack depth per invocation:  0 words
23824
23825
23826 ;          10389 !<BLF/PAGE>
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (108)

```
23828 ;ML4
23829 :
23830
23831 ;          10390 !
23832 ;          10391 BGNTST;
23833 ;          10392
23834 ;          10393 !++
23835 ;          10394 ! TEST NUMBER:  TST 58
23836 ;          10395 !
23837 ;          10396 ! TEST NAME:            ECC HARD ERROR BIT TEST
23838 ;          10397 !
23839 ;          10398 ! TEST DESCRIPTION:
23840 ;          10399 !       VIA ECC_DM AND ECC_DIS TEST THE
23841 ;          10400 !       ECH BIT FOR SETTING AND NOT SETTING BY:
23842 ;          10401 !
23843 ;          10402 !       1.  VIA ECC_DM AND A WRITE TRANSFER FORCE
23844 ;          10403 !           ECC ERRORS INTO THE ML11'S GOOD BLOCK
23845 ;          10404 !
23846 ;          10405 !       2.  WITH ECC_DM = 0 AND ECC_DIS = 0
23847 ;          10406 !           DO A READ TRANSFER AND CHECK ECH SET
23848 ;          10407 !
23849 ;          10408 !       3.  WITH ECC_DM = 0 AND ECC_DIS = 1
23850 ;          10409 !           DO A READ TRANSFER AND CHECK ECH NOT SET
23851 ;          10410 !
23852 ;          10411 ! IMPLICIT INPUTS:
23853 ;          10412 !       IO_BUF
23854 ;          10413 !       A  VECTOR OF 256 WORDS WHERE DATA
23855 ;          10414 !       FOR MBUS READS AND WRITES TRANSFERS
23856 ;          10415 !       IS FOUND.
23857 ;          10416 !
23858 ;          10417 !
23859 ;          10418 !--
23860 ;          10419
23861 ;          10420 local
23862 ;          10421     EXPECTED;                          !EXPECTED ECH_ERR VALUE
23863 ;          10422
23864 ;          10423 CLR_MBUS;                              !CLEAR THE DRIVE
23865 ;          10424 ECC_DM = ONE;                          !SET ECC DIAG MODE
23866 ;          10425 PAR_CRC_WRD = ZEROES;                  !FORCE UNCORRECTABLE ERRORS INTO THE GOOD BLOCK
23867 ;          10426 CRC_A = ZEROES;
23868 ;          10427 CRC_B = ZEROES;
23869 ;          10428 BAI = ONE;
23870 ;          10429 IO_BUF = ONES;
23871 ;          10430 GD_BLK_XFER ();
23872 ;          10431 MLCS1 = write;
23873 ;          10432 TIME_OUT_LOOP;
23874 ;          10433
23875 ;          10434 incr LOOP from 0 to 1 do               !TEST ECH BIT FOR SETTING / NOT SETTING
23876 ;          10435     begin
23877 ;          10436     BGNSUB;                            !START OF SCOPE LOOP
23878 ;          10437     CLR_MBUS;                          !CLEAR THE DRIVE
23879 ;          10438     GD_BLK_XFER ();                    !SET UP A GOOD BLOCK TRANSFER
23880 ;          10439
23881 ;          10440     if .LOOP eql ZERO                  !SELECT THE TEST DATA
23882 ;          10441     then
```

```
 23884 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
 23885 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (108)
 23886
 23887 ;          10442          begin                         !EXPECT ECH BIT TO SET
 23888 ;          10443          ECC_DIS = 0;
 23889 ;          10444          EXPECTED = 1;
 23890 ;          10445          end
 23891 ;          10446      else
 23892 ;          10447          begin                         !EXPECT ECH BIT NOT TO SET
 23893 ;          10448          ECC_DIS = 1;
 23894 ;          10449          EXPECTED = 0;
 23895 ;          10450          end;
 23896 ;          10451
 23897 ;          10452      MLCS1 = read;                     !READ THE ERROR OUT
 23898 ;          10453      TIME_OUT_LOOP;
 23899 ;          10454
 23900 ;          10455      if .ECH_ERR neq .EXPECTED         !IS ECH BIT WHAT WE EXPECTED
 23901 ;          10456      then
 23902 ;          10457          begin                         !ERROR IF NOT
 23903 ;          10458          ERRDF (129, SYNC, DUMPER);
 23904 ;          10459          PRINTB (FOR_FMT, WRD_67, WRD_10, WRD_69, WRD_14);
 23905 ;          10460          PRINTB (FMT_1, .EXPECTED, .ECH_ERR);
 23906 ;          10461          end;
 23907 ;          10462
 23908 ;          10463      ENDSUB;                           !END OF SCOPE LOOP
 23909 ;          10464      end;
 23910 ;          10465
 23911 ;          10466 ENDTST;
 23915
 23916                                          .SBTTL  $T58
 23920 103476 004167 100356          $T58:   JSR     R1,$SAVE3                ;                    1038
 23921 103502 152777 000040 107504           BISB    #40,@ML.REG+40           ;                    1042
 23922 103510 016702 110066                  MOV     ML.DUT,R2
 23923 103514 042702 177770                  BIC     #177770,R2
 23924 103520 142777 000007 107466           BICB    #7,@ML.REG+40
 23925 103526 150277 107462                  BISB    R2,@ML.REG+40
 23926 103532 152777 000001 107534           BISB    #1,@ML.REG+120           ;                    1042
 23927 103540 042777 037400 107556           BIC     #37400,@ML.REG+150       ;                    1042
 23928 103546 142777 000077 107550           BICB    #77,@ML.REG+150          ;                    1042
 23929 103554 142777 000077 107552           BICB    #77,@ML.REG+160          ;                    1042
 23930 103562 152777 000010 107424           BISB    #10,@ML.REG+40           ;                    1042
 23931 103570 012767 177777 106002           MOV     #-1,IO.BUF               ;                    1042
 23932 103576 004767 112266                  JSR     PC,GD.BLK.XFER           ;                    1043
 23933 103602 012777 000061 107344           MOV     #61,@ML.REG              ;                    1043
 23934 103610 105777 107410          1$:     TSTB    @ML.REG+50
 23935 103614 100375                         BPL     1$
 23936 103616 005001                         CLR     R1                       ; LOOP               1043
 23937 103620 104402                 2$:     TRAP    2                        ;                    1043
 23938 103622 152777 000040 107364           BISB    #40,@ML.REG+40           ;                    1043
```

```
23940                                   :ML4                                                    14-May-1981 13:48:54    TOPS
23941                                   ;                                                       14-May-1981 13:05:08    PA:<
23942
23943 103630  016700  107746                    MOV     ML.DUT,R0
23944 103634  042700  177770                    BIC     #177770,R0
23945 103640  142777  000007  107346            BICB    #7,@ML.REG+40
23946 103646  150077  107342                    BISB    R0,@ML.REG+40
23947 103652  004767  112212                    JSR     PC,GD.BLK.XFER          ;                       1043
23948 103656  005701                            TST     R1                      ; LOOP                  1044
23949 103660  001006                            BNE     3$
23950 103662  142777  000002  107404            BICB    #2,@ML.REG+120          ;                       1044
23951 103670  012702  000001                    MOV     #1,R2                   ; *,EXPECTED            1044
23952 103674  000404                            BR      4$                      ;                       1044
23953 103676  152777  000002  107370  3$:       BISB    #2,@ML.REG+120          ;                       1044
23954 103704  005002                            CLR     R2                      ; EXPECTED              1044
23955 103706  012777  000071  107240  4$:       MOV     #71,@ML.REG             ;                       1045
23956 103714  105777  107304          5$:       TSTB    @ML.REG+50
23957 103720  100375                            BPL     5$
23958 103722  010203                            MOV     R2,R3                   ; EXPECTED,*            1045
23959 103724  005000                            CLR     R0
23960 103726  132777  000100  107300            BITB    #100,@ML.REG+60
23961 103734  001401                            BEQ     6$
23962 103736  005200                            INC     R0
23963 103740  020003                  6$:       CMP     R0,R3
23964 103742  001441                            BEQ     8$
23965 103744  104455                            TRAP    55                      ;                       1045
23966 103746  000201                            .WORD   201
23967 103750  010526                            .WORD   SYNC
23968 103752  024060                            .WORD   DUMPER
23969 103754  012746  006442                    MOV     #WRD.14,-(SP)           ;                       1045
23970 103760  012746  007252                    MOV     #WRD.69,-(SP)
23971 103764  012746  006406                    MOV     #WRD.10,-(SP)
23972 103770  012746  007240                    MOV     #WRD.67,-(SP)
23973 103774  012746  006046                    MOV     #FOR.FMT,-(SP)
23974 104000  012746  000005                    MOV     #5,-(SP)
23975 104004  010600                            MOV     SP,R0                   ; SP,*
23976 104006  104414                            TRAP    14
23977 104010  005016                            CLR     (SP)                    ;                       1046
23978 104012  132777  000100  107214            BITB    #100,@ML.REG+60
23979 104020  001401                            BEQ     7$
23980 104022  005216                            INC     (SP)
23981 104024  010246                  7$:       MOV     R2,-(SP)                ; EXPECTED,*
23982 104026  012746  004222                    MOV     #FMT.1,-(SP)
23983 104032  012746  000003                    MOV     #3,-(SP)
23984 104036  010600                            MOV     SP,R0                   ; SP,*
23985 104040  104414                            TRAP    14
23986 104042  062706  000022                    ADD     #22,SP                  ;                       1045
23987 104046  104467                  8$:       TRAP    67                      ;                       1046
23988 104050  006000                            ROR     R0
23989 104052  103662                            BLO     2$
23990 104054  005201                            INC     R1                      ; LOOP                  1043
23991 104056  020127  000001                    CMP     R1,#1                   ; LOOP,*
23992 104062  003656                            BLE     2$
23993 104064  000207                            RTS     PC                      ;                       1038
23994
```

```
23996                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
23997                                   ;                                                       14-May-1981 13:05:08    PA:<
23998
23999                                   ; Routine Size:  124 words
24000                                   ; Maximum stack depth per invocation:  13 words
24005
24006
24010
24011                                           .SBTTL  T58
24015 104066                            T58::
24016 104066  004767  177404            1$:     JSR     PC,$T58                         ;       1046
24017 104072  104466                            TRAP    66
24018 104074  006000                            ROR     R0
24019 104076  103773                            BLO     1$
24020 104100  000207                            RTS     PC
24021
24022                                   ; Routine Size:  6 words
24023                                   ; Maximum stack depth per invocation:  0 words
24028
24029
24030 ;          10467 !<BLF/PAGE>
```

```
24032 ;ML4
24033 ;
24034
24035 ;          10468 !
24036 ;          10469 BGNTST;
24037 ;          10470
24038 ;          10471 !++
24039 ;          10472 ! TEST NUMBER:  TST 59
24040 ;          10473 !
24041 ;          10474 ! TEST NAME:            DATA CHECK ERROR BIT TEST
24042 ;          10475 !
24043 ;          10476 ! TEST DESCRIPTION:
24044 ;          10477 !       VIA ECC_EN, ECC_DM AND ECC_DM TEST THE
24045 ;          10478 !       DCK BIT FOR SETTING AND NOT SETTING BY:
24046 ;          10479 !
24047 ;          10480 !       1.  VIA ECC_DM AND A WRITE TRANSFER FORCE
24048 ;          10481 !           ECC ERRORS INTO THE ML11'S GOOD BLOCK
24049 ;          10482 !
24050 ;          10483 !       2.  WITH DCK_EN = 1, ECC_DM = 0, ECC_DIS = 0
24051 ;          10484 !           DO A READ TRANSFER AND CHECK THE DCK BIT SET
24052 ;          10485 !
24053 ;          10486 !       3.  WITH DCK_EN = 0, ECC_DM = 0, ECC_DIS = 0
24054 ;          10487 !           DO A READ TRANSFER AND CHECK THE DCK BIT NOT SET
24055 ;          10488 !
24056 ;          10489 !       4.  WITH ECK_EN = 1, ECC_DM = 0, ECC_DIS = 1
24057 ;          10490 !           DO A READ TRANSFER AND CHECK THE DCK BIT SET
24058 ;          10491 !
24059 ;          10492 !
24060 ;          10493 ! IMPLICIT INPUTS:
24061 ;          10494 !       IO_BUF
24062 ;          10495 !       A VECTOR OF 256 WORDS WHERE DATA
24063 ;          10496 !       FOR MBUS READS AND WRITES TRANSFERS
24064 ;          10497 !       IS FOUND.
24065 ;          10498 !
24066 ;          10499 !
24067 ;          10500 !--
24068 ;          10501
24069 ;          10502 local
24070 ;          10503     SGL,                          !SINGLE BIT ERROR PARRAMETER
24071 ;          10504     UNC,                          !UNCORRECTABLE ERROR PARRAMETER
24072 ;          10505     EXPECTED;                     !EXPECTED DATA CHECK BIT VALUE
24073 ;          10506
24074 ;          10507 CLR_MBUS;                         !CLEAR THE DRIVE
24075 ;          10508 FOR_ECC_ERR (SGL = 1, UNC = 0);   !FORCE  A SBE
24076 ;          10509 BAI = ONE;
24077 ;          10510 IO_BUF = ZEROES;
24078 ;          10511 GD_BLK_XFER ();                   !SET UP A GOOD BLOCK TRANSFER
24079 ;          10512 MLCS1 = write;                    !WRITE THE DRIVE WITH THE ERROR
24080 ;          10513 TIME_OUT_LOOP;
24081 ;          10514
24082 ;          10515 incr LOOP from 0 to 2 do          !TEST DCK BIT FOR 3 CONDITIONS
24083 ;          10516     begin
24084 ;          10517     BGNSUB;                       !START OF SCOPE LOOP
24085 ;          10518     CLR_MBUS;                     !CLEAR THE DRIVE
24086 ;          10519     GD_BLK_XFER ();               !SET UP A GOOD BLOCK TRANSFER
```

```
24088 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
24089 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (109)
24090
24091 ;          10520
24092 ;          10521      case .LOOP from 0 to 2 of                  !SELECT THE TEST DATA
24093 ;          10522          set
24094 ;          10523
24095 ;          10524          [0] :                                  !EXPECT THE DCK BIT TO SET
24096 ;          10525              begin
24097 ;          10526              DCK_EN = 1;
24098 ;          10527              ECC_DIS = 0;
24099 ;          10528              EXPECTED = 1;
24100 ;          10529              end;
24101 ;          10530
24102 ;          10531          [1] :                                  !EXPECT THE DCK BIT NOT TO SET
24103 ;          10532              begin
24104 ;          10533              DCK_EN = 0;
24105 ;          10534              ECC_DIS = 0;
24106 ;          10535              EXPECTED = 0;
24107 ;          10536              end;
24108 ;          10537
24109 ;          10538          [2] :                                  !EXPECT THE DCK BIT TO SET
24110 ;          10539              begin
24111 ;          10540              DCK_EN = 1;
24112 ;          10541              ECC_DIS = 1;
24113 ;          10542              EXPECTED = 1;
24114 ;          10543              end;
24115 ;          10544          tes;
24116 ;          10545
24117 ;          10546      MLCS1 = read;                              !READ THE ERROR AND CLOCK THE DCK BIT
24118 ;          10547      TIME_OUT_LOOP;
24119 ;          10548
24120 ;          10549      if .DCK neq .EXPECTED                      !IS THE DCK BIT WHAT WE EXPECTED
24121 ;          10550      then
24122 ;          10551          begin                                  !REPORT THE ERRRO IF NOT THE SAME
24123 ;          10552
24124 ;          10553          if .LOOP neq ONE then ERRDF (130, SYNC, DUMPER) else ERRDF (130, ASYNC, DUMPER);
24125 ;          10554
24126 ;          10555                                                 !CHOOSE WHICH THE ERROR IS ON
24127 ;          10556          PRINTB (THR_FMT, WRD_72, PHR_5, WRD_10);
24128 ;          10557          PRINTB (FMT_1, .EXPECTED, .DCK);
24129 ;          10558          end;
24130 ;          10559
24131 ;          10560      ENDSUB;                                    !END OF THE SCOPE LOOP
24132 ;          10561      end;
24133 ;          10562
24134 ;          10563 ENDTST;
24138
24139                                  .SBTTL  $T59
```

```
24144                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
24145                                   ;                                                       14-May-1981 13:05:08    PA:<
24146
24147 104102  004167  077752            $T59:   JSR     R1,$SAVE3                       ;                               1046
24148 104106  152777  000040  107100            BISB    #40,@ML.REG+40                  ;                               1050
24149 104114  016702  107462                    MOV     ML.DUT,R2
24150 104120  042702  177770                    BIC     #177770,R2
24151 104124  142777  000007  107062            BICB    #7,@ML.REG+40
24152 104132  150277  107056                    BISB    R2,@ML.REG+40
24153 104136  012701  000001                    MOV     #1,R1                           ; *,SGL                         1050
24154 104142  010146                            MOV     R1,-(SP)                        ; SGL,*
24155 104144  005001                            CLR     R1                              ; UNC
24156 104146  005046                            CLR     -(SP)
24157 104150  004767  111604                    JSR     PC,FOR.ECC.ERR
24158 104154  152777  000010  107032            BISB    #10,@ML.REG+40                  ;                               1050
24159 104162  005067  105412                    CLR     IO.BUF                          ;                               1051
24160 104166  004767  111676                    JSR     PC,GD.BLK.XFER                  ;                               1051
24161 104172  012777  000061  106754            MOV     #61,@ML.REG                     ;                               1051
24162 104200  105777  107020            1$:     TSTB    @ML.REG+50
24163 104204  100375                            BPL     1$
24164 104206  005001                            CLR     R1                              ; LOOP                          1051
24165 104210  104402            2$:     TRAP    2                               ;                               1051
24166 104212  152777  000040  106774            BISB    #40,@ML.REG+40                  ;                               1051
24167 104220  016700  107356                    MOV     ML.DUT,R0
24168 104224  042700  177770                    BIC     #177770,R0
24169 104230  142777  000007  106756            BICB    #7,@ML.REG+40
24170 104236  150077  106752                    BISB    R0,@ML.REG+40
24171 104242  004767  111622                    JSR     PC,GD.BLK.XFER                  ;                               1051
24172 104246  010100                            MOV     R1,R0                           ; LOOP,*                        1052
24173 104250  006300                            ASL     R0
24174 104252  066007  104256                    ADD     3$(R0),PC
24175 104256  000006            3$:     .WORD   4$-3$
24176 104260  000024                    .WORD   5$-3$
24177 104262  000044                    .WORD   6$-3$
24178 104264  152777  000004  107002    4$:     BISB    #4,@ML.REG+120                  ;                               1052
24179 104272  142777  000002  106774            BICB    #2,@ML.REG+120                  ;                               1052
24180 104300  000416                            BR      7$                              ;                               1052
24181 104302  142777  000004  106764    5$:     BICB    #4,@ML.REG+120                  ;                               1053
24182 104310  142777  000002  106756            BICB    #2,@ML.REG+120                  ;                               1053
24183 104316  005002                            CLR     R2                              ; EXPECTED                      1053
24184 104320  000410                            BR      8$                              ;                               1052
24185 104322  152777  000004  106744    6$:     BISB    #4,@ML.REG+120                  ;                               1054
24186 104330  152777  000002  106736            BISB    #2,@ML.REG+120                  ;                               1054
24187 104336  012702  000001            7$:     MOV     #1,R2                           ; *,EXPECTED                    1054
24188 104342  012777  000071  106604    8$:     MOV     #71,@ML.REG                     ;                               1054
24189 104350  105777  106650            9$:     TSTB    @ML.REG+50
24190 104354  100375                            BPL     9$
24191 104356  010203                            MOV     R2,R3                           ; EXPECTED,*                    1054
24192 104360  005000                            CLR     R0
24193 104362  005777  106646                    TST     @ML.REG+60
24194 104366  100001                            BPL     10$
24195 104370  005200                            INC     R0
24196 104372  020003            10$:    CMP     R0,R3
24197 104374  001446                            BEQ     14$
24198 104376  020127  000001                    CMP     R1,#1                           ; LOOP,*                        1055
```

```
24200                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
24201                                   ;                                                       14-May-1981 13:05:08    PA:<
24202
24203 104402  001405                            BEQ     11$
24204 104404  104455                            TRAP    55
24205 104406  000202                            .WORD   202
24206 104410  010526                            .WORD   SYNC
24207 104412  024060                            .WORD   DUMPER
24208 104414  000404                            BR      12$
24209 104416  104455                    11$:    TRAP    55
24210 104420  000202                            .WORD   202
24211 104422  010464                            .WORD   ASYNC
24212 104424  024060                            .WORD   DUMPER
24213 104426  012746  006406            12$:    MOV     #WRD.10,-(SP)                   ;                       1055
24214 104432  012746  007472                    MOV     #PHR.5,-(SP)
24215 104436  012746  007302                    MOV     #WRD.72,-(SP)
24216 104442  012746  006034                    MOV     #THR.FMT,-(SP)
24217 104446  012746  000004                    MOV     #4,-(SP)
24218 104452  010600                            MOV     SP,R0                           ; SP,*
24219 104454  104414                            TRAP    14
24220 104456  005016                            CLR     (SP)                            ;                       1055
24221 104460  005777  106550                    TST     @ML.REG+60
24222 104464  100001                            BPL     13$
24223 104466  005216                            INC     (SP)
24224 104470  010246                    13$:    MOV     R2,-(SP)                        ; EXPECTED,*
24225 104472  012746  004222                    MOV     #FMT.1,-(SP)
24226 104476  012746  000003                    MOV     #3,-(SP)
24227 104502  010600                            MOV     SP,R0                           ; SP,*
24228 104504  104414                            TRAP    14
24229 104506  062706  000020                    ADD     #20,SP                          ;                       1055
24230 104512  104467                    14$:    TRAP    67                              ;                       1055
24231 104514  006000                            ROR     R0
24232 104516  103634                            BLO     2$
24233 104520  005201                            INC     R1                              ; LOOP                  1051
24234 104522  020127  000002                    CMP     R1,#2                           ; LOOP,*
24235 104526  003630                            BLE     2$
24236 104530  022626                            CMP     (SP)+,(SP)+                     ;                       1046
24237 104532  000207                            RTS     PC
24238
24239                                   ; Routine Size:  141 words
24240                                   ; Maximum stack depth per invocation:  14 words
24245
24246
24250
24251                                           .SBTTL  T59
```

```
24256                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
24257                                   ;                                                       14-May-1981 13:05:08    PA:<
24258
24259 104534                            T59::
24260 104534  004767  177342            1$:     JSR     PC,$T59                 ;                               1056
24261 104540  104466                            TRAP    66
24262 104542  006000                            ROR     R0
24263 104544  103773                            BLO     1$
24264 104546  000207                            RTS     PC
24265
24266                                   ; Routine Size:  6 words
24267                                   ; Maximum stack depth per invocation:  0 words
24272
24273
24274 ;          10564 !<BLF/PAGE>
```

14-May-1981 13:48:54 TOPS-20 Bliss-16 V2(212)
14-May-1981 13:05:08 PA:<NEALE>ML4.BLI.7 (110)

```
24276 ;ML4
24277 ;
24278
24279 ;          10565 !
24280 ;          10566 BGNTST;
24281 ;          10567
24282 ;          10568 !++
24283 ;          10569 ! TEST NUMBER:  TST 60
24284 ;          10570 !
24285 ;          10571 ! TEST NAME:    PROM DATA TEST
24286 ;          10572 !
24287 ;          10573 ! TEST DESCRIPTION:
24288 ;          10574 !
24289 ;          10575 !       VERIFY THAT CHECK SUM VALUES FOR
24290 ;          10576 !       ALL PRESENT MEMORY ARRAY UV PROM
24291 ;          10577 !       LOCATIONS ARE ERROR FREE BY:
24292 ;          10578 !
24293 ;          10579 !       1.  DOING MBUS TRANSFERS
24294 ;          10580 !           AT ALL PRESENT BLOCKS
24295 ;          10581 !           AND TESTING THE UNS BIT
24296 ;          10582 !           TO BE CLEARED.
24297 ;          10583 !
24298 ;          10584 ! IMPLICIT INPUTS:
24299 ;          10585 !
24300 ;          10586 !       IO_BUF
24301 ;          10587 !
24302 ;          10588 !       A VECTOR OF 256 WORDS WHERE
24303 ;          10589 !       DATA FOR MBUS READ AND WRITE
24304 ;          10590 !       FUNCTIONS IS FOUND.
24305 ;          10591 !
24306 ;          10592 !
24307 ;          10593 !       CHIP_SIZ
24308 ;          10594 !       INDICATED THE SIZE OF THE
24309 ;          10595 !       ARRAY MODULES MOS RAMS.
24310 ;          10596 !
24311 ;          10597 !
24312 ;          10598 !--
24313 ;          10599
24314 ;          10600 incr DSA_CNT from 0 to .LST_BLK do          !WRITE TO ALL PRESENT BLK'S AND CHECK UNS BIT
24315 ;          10601     begin
24316 ;          10602     BGNSUB;
24317 ;          10603     CLR_MBUS;
24318 ;          10604     ECC_DIS = ONE;                         !DISABLE ECC
24319 ;          10605     MLWC = not 255;                        !LOAD WORD COUNT
24320 ;          10606     MLBA = IO_BUF;                         !LOAD UBUS ADRS
24321 ;          10607     MLDA = .DSA_CNT;                       !LOAD DSA
24322 ;          10608     MLCS1 = write;                         !DO A WRITE FUNCTION
24323 ;          10609     TIME_OUT_LOOP;
24324 ;          10610
24325 ;          10611     if .UNS IS_SET                         !SEE IF XFERR CAUSED AN UNS ERROR
24326 ;          10612     then
24327 ;          10613         begin                              !ERROR IF SET
24328 ;          10614         ERRDF (107, ARR_DAT, DUMPER);
24329 ;          10615         PRINTB (TWO_FMT, WRD_35, PHR_4);
24330 ;          10616         PRINTB (FMT_9, .DSA_CNT);
```

```
24332 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
24333 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (110)
24334
24335 ;          10617
24336 ;          10618          if .CHIP_SIZ eql 64                    !NEED TO KNOW CHIP SIZE TO PRINT ARRAY NUMBER
24337 ;          10619          then
24338 ;          10620              begin
24339 ;          10621              PRINTB (FMT_8, ((.DSA_CNT<11, 4>) + 1));     !64K MOS RAM
24340 ;          10622              end
24341 ;          10623          else
24342 ;          10624              begin
24343 ;          10625              PRINTB (FMT_8, ((.DSA_CNT<9, 4>) + 1));      !16K MOS RAM
24344 ;          10626              end
24345 ;          10627
24346 ;          10628          end;
24347 ;          10629
24348 ;          10630      ENDSUB;
24349 ;          10631      end;
24350 ;          10632
24351 ;          10633 ENDTST;
24355
24356                                      .SBTTL  $T60
24360 104550  004167  077270        $T60: JSR     R1,$SAVE2                ;                              1056
24361 104554  016702  105012              MOV     LST.BLK,R2               ;                              1060
24362 104560  005001                      CLR     R1                       ; DSA.CNT
24363 104562  000536                      BR      6$
24364 104564  104402                1$:   TRAP    2                        ;                              1060
24365 104566  152777  000040  106420      BISB    #40,@ML.REG+40           ;                              1060
24366 104574  016700  107002              MOV     ML.DUT,R0
24367 104600  042700  177770              BIC     #177770,R0
24368 104604  142777  000007  106402      BICB    #7,@ML.REG+40
24369 104612  150077  106376              BISB    R0,@ML.REG+40
24370 104616  152777  000002  106450      BISB    #2,@ML.REG+120           ;                              1060
24371 104624  012777  177400  106332      MOV     #-400,@ML.REG+10         ;                              1060
24372 104632  012777  011600  106334      MOV     #IO.BUF,@ML.REG+20       ;                              1060
24373 104640  010177  106340              MOV     R1,@ML.REG+30            ; DSA.CNT,*                    1060
24374 104644  012777  000061  106302      MOV     #61,@ML.REG              ;                              1060
24375 104652  105777  106346          2$: TSTB    @ML.REG+50
24376 104656  100375                      BPL     2$
24377 104660  032777  040000  106346      BIT     #40000,@ML.REG+60        ;                              1061
24378 104666  001470                      BEQ     5$
24379 104670  104455                      TRAP    55                       ;                              1061
24380 104672  000153                      .WORD   153
24381 104674  010570                      .WORD   ARR.DAT
24382 104676  024060                      .WORD   DUMPER
24383 104700  012746  007454              MOV     #PHR.4,-(SP)             ;                              1061
24384 104704  012746  006666              MOV     #WRD.35,-(SP)
24385 104710  012746  006024              MOV     #TWO.FMT,-(SP)
24386 104714  012746  000003              MOV     #3,-(SP)
```

```
                                        ;ML4                                                    14-May-1981 13:48:54    TOPS
                                        .                                                       14-May-1981 13:05:08    PA:<
24391 104720 010600                             MOV     SP,R0                   ; SP,*
24392 104722 104414                             TRAP    14
24393 104724 010116                             MOV     R1,(SP)                 ; DSA.CNT,*                             1061
24394 104726 012746 004644                      MOV     #FMT.9,-(SP)
24395 104732 012746 000002                      MOV     #2,-(SP)
24396 104736 010600                             MOV     SP,R0                   ; SP,*
24397 104740 104414                             TRAP    14
24398 104742 026727 104622 000100               CMP     CHIP.SIZ,#100           ;                                       1061
24399 104750 001020                             BNE     3$
24400 104752 010100                             MOV     R1,R0                   ; DSA.CNT,*                             1062
24401 104754 006200                             ASR     R0
24402 104756 006200                             ASR     R0
24403 104760 006200                             ASR     R0
24404 104762 000300                             SWAB    R0
24405 104764 042700 177760                      BIC     #177760,R0
24406 104770 010046                             MOV     R0,-(SP)
24407 104772 005216                             INC     (SP)
24408 104774 012746 004610                      MOV     #FMT.8,-(SP)
24409 105000 012746 000002                      MOV     #2,-(SP)
24410 105004 010600                             MOV     SP,R0                   ; SP,*
24411 105006 104414                             TRAP    14
24412 105010 000415                             BR      4$                      ;                                       1061
24413 105012 010100                     3$:     MOV     R1,R0                   ; DSA.CNT,*                             1062
24414 105014 006200                             ASR     R0
24415 105016 000300                             SWAB    R0
24416 105020 042700 177760                      BIC     #177760,R0
24417 105024 010046                             MOV     R0,-(SP)
24418 105026 005216                             INC     (SP)
24419 105030 012746 004610                      MOV     #FMT.8,-(SP)
24420 105034 012746 000002                      MOV     #2,-(SP)
24421 105040 010600                             MOV     SP,R0                   ; SP,*
24422 105042 104414                             TRAP    14
24423 105044 062706 000022              4$:     ADD     #22,SP                  ;                                       1061
24424 105050 104467                     5$:     TRAP    67                      ;                                       1062
24425 105052 006000                             ROR     R0
24426 105054 103643                             BLO     1$
24427 105056 005201                             INC     R1                      ; DSA.CNT                               1060
24428 105060 020102                     6$:     CMP     R1,R2                   ; DSA.CNT,*
24429 105062 003640                             BLE     1$
24430 105064 000207                             RTS     PC                      ;                                       1056
24431
24432                                   ; Routine Size:  103 words
24433                                   ; Maximum stack depth per invocation:  12 words
24438
24439
```

```
24444                                   ;ML4                                                    14-May-1981 13:48:54    TOPS
24445                                   ;                                                       14-May-1981 13:05:08    PA:<
24446
24447
24448                                           .SBTTL  T60
24452 105066                            T60::
24453 105066  004767  177456            1$:     JSR     PC,$T60                 ;                                       1063
24454 105072  104466                            TRAP    66
24455 105074  006000                            ROR     R0
24456 105076  103773                            BLO     1$
24457 105100  000207                            RTS     PC
24458
24459                                   ; Routine Size:  6 words
24460                                   ; Maximum stack depth per invocation:  0 words
24465
24466
24467 ;          10634 !<BLF/PAGE>
```

```
24469 ;ML4                                                    14-May-1981 13:48:54    TOPS-20 Bliss-16 V2(212)
24470 ;                                                       14-May-1981 13:05:08    PA:<NEALE>ML4.BLI.7 (111)
24471
24472 ;          10635 !       THE CLEANUP CODING SECTION IS EXECUTED AFTER THE
24473 ;          10636 !       HARDWARE TESTS ARE RUN ON A LOGICAL UNIT.
24474 ;          10637 BGNCLN;
24475 ;          10638 CLR = ONE;                                !CLEAR THE MASS BUS
24476 ;          10639 return;
24477 ;          10640 ENDCLN;
24481
24482                                        .SBTTL  LCLEAN
24486 105102  152777  000040  106104  LCLEAN: BISB    #40,@ML.REG+40                  ;                1063
24487 105110  000207                          RTS     PC                              ;                1063
24488
24489                                  ; Routine Size:  4 words
24490                                  ; Maximum stack depth per invocation:  0 words
24495
24496
24500
24501                                        .SBTTL  L$CLEAN
24505 105112                           L$CLEAN::
24506 105112  004767  177764                  JSR     PC,LCLEAN                       ;                1063
24507 105116  104412                          TRAP    12
24508 105120  000207                          RTS     PC
24509
24510                                  ; Routine Size:  4 words
24511                                  ; Maximum stack depth per invocation:  0 words
24516
24517
24518 ;          10641 LASTAD;
24519 ;          10642 BGNSETUP (0);
24520 ;          10643 ENDSETUP;
```

```
                                ;ML4                                                    14-May-1981 13:48:54    TOPS
                                ;                                                       14-May-1981 13:05:08    PA:<

105122  105126                BL$LAS::.WORD   T$FREE
105124  000000                        .WORD   <<T$FREE-<BL$LAS+4>>/2>
105126  000000                T$FREE::.WORD   0

        105126                L$LAST==            BL$LAS+4
        000000                T$PTHV==            0

                                        .SBTTL  $END.LINK
105130                          $END.LINK::
105130  000207                        RTS     PC                              ;       1064

                                ; Routine Size:  1 word
                                ; Maximum stack depth per invocation:  0 words

;         10644 end
;         10645
;         10646 eludom

                                ; OTS external references
                                        .GLOBL  BL$GT2, $SAVE5, $SAVE4, $SAVE3
                                        .GLOBL  $SAVE2, BL$PU2, BL$PU1, BL$GT1
                                        .GLOBL  BL$DIV, BL$MOD, BL$MUL

                                ; Size:         14699 code + 1916 data words
                                ; Run Time:     03:15.0
                                ; Elapsed Time: 09:39.2
                                ; Memory Used:  118 pages
                                ; Compilation Complete

000001                                  .END
```

```
ADR   = 000020 G    C$CLP1= 000006    EF.RES= 000037 G    FNC.3 = 007750    IER   = 020000 G
ARR.DA= 010570      C$CVEC= 000036    EF.STA= 000040 G    FNC.4 = 007756    INTER = 010672
ARR.IN  011562      C$DCLN= 000044    EIG.FM= 006142      FNC.5 = 007774    IO.BUF  011600
ARR.16  011574      C$DODU= 000051    ELV.FM= 006244      FNC.6 = 010004    ISR   = 000100 G
ASSEMB= 000010      C$DRPT= 000024    ERRBLK  002160 G    FNC.7 = 010012    IXE   = 004000 G
ASYNC = 010464      C$DU  = 000053    ERRMSG  002156 G    FNC.8 = 010022    I$AU  = 000041
A.CAL   013140      C$EDIT= 000003    ERRNBR  002154 G    FNC.9 = 010036    I$AUTO= 000041
A.GEN   013146      C$ERDF= 000055    ERRTYP  002152 G    FORCE.  015602    I$CLN = 000041
BIT0  = 000001 G    C$ERHR= 000056    ERR.CH  015212      FOR.EC  015760    I$DU  = 000041
BIT00 = 000001 G    C$ERRO= 000060    EVL   = 000004 G    FOR.FM= 006046    I$HRD = 000041
BIT01 = 000002 G    C$ERSF= 000054    E$END = 002100      F$AU  = 000015    I$INIT= 000041
BIT02 = 000004 G    C$ERSO= 000057    E$LOAD= 000035      F$AUTO= 000020    I$MOD = 000041
BIT03 = 000010 G    C$ESCA= 000010    E2.TEM= 011156      F$BGN = 000040    I$MSG = 000041
BIT04 = 000020 G    C$ESEG= 000005    FIND.C  015430      F$CLEA= 000007    I$PROT= 000040
BIT05 = 000040 G    C$ESUB= 000003    FIRST.  016040      F$DU  = 000016    I$PTAB= 000041
BIT06 = 000100 G    C$ETST= 000001    FIV.FM= 006062      F$END = 000041    I$PWR = 000041
BIT07 = 000200 G    C$EXIT= 000032    FMT.1 = 004222      F$HARD= 000004    I$RPT = 000041
BIT08 = 000400 G    C$GETB= 000026    FMT.10= 004676      F$HW  = 000013    I$SEG = 000041
BIT09 = 001000 G    C$GETW= 000027    FMT.11= 004750      F$INIT= 000006    I$SETU= 000041
BIT1  = 000002 G    C$GMAN= 000043    FMT.12= 005004      F$JMP = 000050    I$SFT = 000041
BIT10 = 002000 G    C$GPHR= 000042    FMT.13= 005034      F$MOD = 000000    I$SRV = 000041
BIT11 = 004000 G    C$GPLO= 000030    FMT.14= 005110      F$MSG = 000011    I$SUB = 000041
BIT12 = 010000 G    C$GPRI= 000040    FMT.15= 005150      F$PROT= 000021    I$TST = 000041
BIT13 = 020000 G    C$INIT= 000011    FMT.16= 005216      F$PWR = 000017    J$JMP = 000167
BIT14 = 040000 G    C$INLP= 000020    FMT.17= 005306      F$RPT = 000012    LAST.B  016122
BIT15 = 100000 G    C$MANI= 000050    FMT.18= 005342      F$SEG = 000003    LAU     004210
BIT2  = 000004 G    C$MEM = 000031    FMT.19= 005412      F$SOFT= 000005    LAUTO   004164
BIT3  = 000010 G    C$MSG = 000023    FMT.2 = 004266      F$SRV = 000010    LCLEAN  105102
BIT4  = 000020 G    C$OPEN= 000034    FMT.20= 005462      F$SUB = 000002    LDU     004176
BIT5  = 000040 G    C$PNTB= 000014    FMT.21= 005534      F$SW  = 000014    LD.LNG  017132
BIT6  = 000100 G    C$PNTF= 000017    FMT.22= 005606      F$TEST= 000001    LINIT   024722
BIT7  = 000200 G    C$PNTS= 000016    FMT.23= 005652      GD.BLK  016070    LOAD.S  013604
BIT8  = 000400 G    C$PNTX= 000015    FMT.24= 005722      GOOD.B  011564    LOE   = 040000 G
BIT9  = 001000 G    C$QIO = 000377    FMT.25= 005746      G$CNTO= 000200    LOT   = 000010 G
BL$DIV  003752 G    C$RDBU= 000007    FMT.26= 005764      G$DELM= 000372    LRPT    004152
BL$GT1  003016 G    C$REFG= 000047    FMT.3 = 004354      G$DISP= 000003    LST.AR  011576
BL$GT2  003140 G    C$RESE= 000033    FMT.4 = 004400      G$EXCP= 000400    LST.BL  011572
BL$LAS  105122 G    C$REVI= 000003    FMT.5 = 004430      G$HILI= 000002    LST.DU  013134
BL$MOD  003764 G    C$RFLA= 000021    FMT.6 = 004532      G$LOLI= 000001    L$ACP   002110 G
BL$MUL  003526 G    C$RPT = 000025    FMT.7 = 004562      G$NO  = 000000    L$APT   002036 G
BL$PU1  003302 G    C$SEFG= 000046    FMT.8 = 004610      G$OFFS= 000400    L$AU    004212 G
BL$PU2  003376 G    C$SPRI= 000041    FMT.9 = 004644      G$OFSI= 000376    L$AUT   002070 G
BL$SHF  003776 G    C$SVEC= 000037    FNC.1 = 007726      G$PRMA= 000001    L$AUTO  004166 G
BOE   = 000400 G    C$TPRI= 000013    FNC.10= 010050      G$PRMD= 000002    L$CCP   002106 G
B.CAL   013142      DATA.L= 010732    FNC.11= 010060      G$PRML= 000000    L$CLEA  105112 G
B.GEN   013150      DAT.DM  014162    FNC.12= 010100      G$RADA= 000140    L$CO    002032 G
CAL.CR  014612      DFPTBL  002356 G  FNC.13= 010112      G$RADB= 000000    L$DEPO  002011 G
CHIP.S  011570      DIAGMC= 000000    FNC.14= 010122      G$RADD= 000040    L$DESC  002130 G
C$AU  = 000052      DIVMOD  003570    FNC.15= 010136      G$RADL= 000120    L$DESP  002076 G
C$AUTO= 000061      DRIVE.  013132    FNC.16= 010150      G$RADO= 000020    L$DEVP  002060 G
C$BRK = 000022      DT.1    013534    FNC.17= 010162      G$XFER= 000004    L$DISP  002164 G
C$BSEG= 000004      DUMPER  024060 G  FNC.18= 010172      G$YES = 000010    L$DLY   002116 G
C$BSUB= 000002      D1.TEM= 011152    FNC.19= 010204      HELP  = 000000    L$DTP   002040 G
C$CEFG= 000045      D2.TEM= 011154    FNC.2 = 007742      HOE   = 100000 G  L$DTYP  002034 G
C$CLCK= 000062      EF.CON= 000036 G  FNC.21= 010216      HW.OR.  011160    L$DU    004200 G
C$CLEA= 000012      EF.NEW= 000035 G  FNC.22= 010230      IBE   = 010000 G  L$DUT   002072 G
C$CLOS= 000035      EF.PWR= 000034 G  FNC.23= 010242      IDU   = 000040 G  L$DVTY  002122 G
```

```
L$EF    002052 G    O$APTS= 000001      P.AAT   005462    P.ACY   006732    P.AFD   010012
L$ENVI  002044 G    O$AU  = 000001      P.AAU   005534    P.ACZ   006746    P.AFE   010022
L$ERRT  002152 G    O$BGNR= 000001      P.AAV   005606    P.ADA   006754    P.AFF   010036
L$ETP   002102 G    O$BGNS= 000001      P.AAW   005652    P.ADB   006764    P.AFG   010050
L$EXP1  002046 G    O$DU  = 000001      P.AAX   005722    P.ADC   006772    P.AFH   010060
L$EXP4  002064 G    O$ERRT= 000001      P.AAY   005746    P.ADD   007004    P.AFI   010100
L$EXP5  002066 G    O$GNSW= 000001      P.AAZ   005764    P.ADE   007016    P.AFJ   010112
L$HARD  002402 G    O$PCIN= 000001      P.ABA   006016    P.ADF   007026    P.AFK   010122
L$HIME  002120 G    O$SETU= 000001      P.ABB   006024    P.ADG   007036    P.AFL   010136
L$HPCP  002016 G    PAR.DI  011566      P.ABC   006034    P.ADH   007046    P.AFM   010150
L$HPTP  002022 G    PD.TEM  013120      P.ABD   006046    P.ADI   007054    P.AFN   010162
L$HW    002356 G    PHR.1 = 007366      P.ABE   006062    P.ADJ   007066    P.AFO   010172
L$ICP   002104 G    PHR.10= 007564      P.ABF   006100    P.ADK   007074    P.AFP   010204
L$INIT  025666 G    PHR.11= 007576      P.ABG   006120    P.ADL   007104    P.AFQ   010216
L$LADP  002026 G    PHR.12= 007614      P.ABH   006142    P.ADM   007114    P.AFR   010230
L$LAST= 105126 G    PHR.13= 007636      P.ABI   006166    P.ADN   007122    P.AFS   010242
L$LOAD  002100 G    PHR.14= 007662      P.ABJ   006214    P.ADO   007130    P.AFT   010256
L$LUN   002074 G    PHR.15= 007714      P.ABK   006244    P.ADP   007140    P.AFU   010266
L$MREV  002050 G    PHR.2 = 007404      P.ABL   006276    P.ADQ   007152    P.AFV   010274
L$NAME  002000 G    PHR.3 = 007422      P.ABM   006302    P.ADR   007156    P.AFW   010302
L$PRIO  002042 G    PHR.4 = 007454      P.ABN   006314    P.ADS   007174    P.AFX   010310
L$PROT  002750 G    PHR.5 = 007472      P.ABO   006322    P.ADT   007210    P.AFY   010316
L$PRT   002112 G    PHR.6 = 007504      P.ABP   006330    P.ADU   007216    P.AFZ   010324
L$REPP  002062 G    PHR.7 = 007516      P.ABQ   006336    P.ADV   007224    P.AGA   010332
L$REV   002010 G    PHR.8 = 007540      P.ABR   006344    P.ADW   007232    P.AGB   010340
L$RPT   004154 G    PHR.9 = 007552      P.ABS   006360    P.ADX   007240    P.AGC   010346
L$SOFT  002720 G    PNT   = 001000 G    P.ABT   006374    P.ADY   007246    P.AGD   010354
L$SPC   002056 G    PRI   = 002000 G    P.ABU   006406    P.ADZ   007252    P.AGE   010362
L$SPCP  002020 G    PRI00 = 000000 G    P.ABV   006416    P.AEA   007260    P.AGF   010370
L$SPTP  002024 G    PRI01 = 000040 G    P.ABW   006426    P.AEB   007272    P.AGG   010376
L$STA   002030 G    PRI02 = 000100 G    P.ABX   006436    P.AEC   007302    P.AGH   010404
L$SW    002376 G    PRI03 = 000140 G    P.ABY   006442    P.AED   007310    P.AGI   010412
L$TEST  002114 G    PRI04 = 000200 G    P.ABZ   006454    P.AEE   007316    P.AGJ   010420
L$TIML  002014 G    PRI05 = 000240 G    P.ACA   006462    P.AEF   007330    P.AGK   010430
L$UNIT  002012 G    PRI06 = 000300 G    P.ACB   006470    P.AEG   007340    P.AGL   010436
L10000  002374      PRI07 = 000340 G    P.ACC   006502    P.AEH   007346    P.AGM   010444
L10001  002400      PRSN    002376 G    P.ACD   006506    P.AEI   007366    P.AGN   010454
L10002  002476      PTBL.P  011556      P.ACE   006514    P.AEJ   007404    P.AGO   010464
L10003  002726      P.AAA   004222      P.ACF   006522    P.AEK   007422    P.AGP   010526
MEM.AR= 010630      P.AAB   004266      P.ACG   006530    P.AEL   007454    P.AGQ   010570
ML.DUT  013602      P.AAC   004354      P.ACH   006544    P.AEM   007472    P.AGR   010630
ML.LUN  013600      P.AAD   004400      P.ACI   006552    P.AEN   007504    P.AGS   010672
ML.REG  013154      P.AAE   004430      P.ACJ   006560    P.AEO   007516    P.AGT   010732
MSGH1   002476      P.AAF   004532      P.ACK   006574    P.AEP   007540    P.AGU   010772
MSGH2   002512      P.AAG   004562      P.ACL   006602    P.AEQ   007552    P.AGV   011026
MSGH3   002540      P.AAH   004610      P.ACM   006610    P.AER   007564    P.AGW   011060
MSGH4   002563      P.AAI   004644      P.ACN   006624    P.AES   007576    P.AGX   011106
MSGH5   002614      P.AAJ   004676      P.ACO   006632    P.AET   007614    P.CAL   013144
MSGH6   002646      P.AAK   004750      P.ACP   006646    P.AEU   007636    P.GEN   013152
MSGH7   002672      P.AAL   005004      P.ACQ   006654    P.AEV   007662    RAS.IN  013124
MSGS1   002726      P.AAM   005034      P.ACR   006660    P.AEW   007714    RD.CS1  017540
MSDUMP  024070      P.AAN   005110      P.ACS   006666    P.AEX   007726    RD.DA   020074
NIB.SA  011152      P.AAO   005150      P.ACT   006674    P.AEY   007742    RD.DAT  013130
NIN.FM= 006166      P.AAP   005216      P.ACU   006700    P.AEZ   007750    RD.DS   022774
ONEFIL= 000001      P.AAQ   005306      P.ACV   006710    P.AFA   007756    RD.D1   021662
ONE.FM= 006016      P.AAR   005342      P.ACW   006716    P.AFB   007774    RD.D2   022174
OP.NUM  011560      P.AAS   005412      P.ACX   006726    P.AFC   010004    RD.D3   022522
```

```
RD.EE   021536          TST.LN  016154          T33     060514 G        WRD.30= 006624          WRT.D2  022110
RD.EL   021474          TWO.FM= 006024          T34     061450 G        WRD.31= 006632          WRT.D3  022422
RD.ER   017716          T$ARGC= 000002          T35     062102 G        WRD.32= 006646          WRT.EE  021534
RD.E1   020652          T$CODE= 000130          T36     063140 G        WRD.33= 006654          WRT.EL  021472
RD.E2   021072          T$ERRN= 000000          T37     064046 G        WRD.34= 006660          WRT.ER  017646
RD.MR   020252          T$EXCP= 000000          T38     065052 G        WRD.35= 006666          WRT.E1  020566
RD.PA   020444          T$FREE  105126 G        T39     065500 G        WRD.36= 006674          WRT.E2  020774
RD.PD   021334          T$GMAN= 000000          T4      027230 G        WRD.37= 006700          WRT.MR  020202
RD.REG  023470          T$HILI= 000007          T40     066466 G        WRD.38= 006710          WRT.PA  020360
RFG.IN  013136          T$LAST= 000000          T41     067520 G        WRD.39= 006716          WRT.PD  021226
REG.1 = 010256          T$LOLI= 000000          T42     070272 G        WRD.4 = 006322          WRT.RE  023102
REG.10= 010346          T$LSYM= 010000          T43     071226 G        WRD.40= 006726          WT.DAT  013126
REG.11= 010354          T$NEST= 177777          T44     071436 G        WRD.41= 006732          W.C.SI  013122
REG.12= 010362          T$NS0 = 000000          T45     071622 G        WRD.42= 006746          XOR.LN  016650
REG.13= 010370          T$NS1 = 000021          T46     072320 G        WRD.43= 006754          X$ALWA= 000000
REG.14= 010376          T$PTHV= 000000 G        T47     072612 G        WRD.44= 006764          X$FALS= 000040
REG.15= 010404          T$PTNU= 000000          T48     073476 G        WRD.45= 006772          X$OFFS= 000400
REG.16= 010412          T$SAVL= 177777          T49     074202 G        WRD.46= 007004          X$TRUE= 000020
REG.17= 010420          T$SEGL= 177777          T5      027652 G        WRD.47= 007016          $END.L  105130 G
REG.18= 010430          T$SUBN= 000000          T50     074620 G        WRD.48= 007026          $PATCH  002756 G
REG.19= 010436          T$TAGL= 177777          T51     075672 G        WRD.49= 007036          $SAVE2  004044 G
REG.2 = 010266          T$TAGN= 010005          T52     076444 G        WRD.5 = 006330          $SAVE3  004060 G
REG.20= 010444          T$TEMP= 000000          T53     077244 G        WRD.50= 007046          $SAVE4  004076 G
REG.21= 010454          T$TEST= 000000          T54     100266 G        WRD.51= 007054          $SAVE5  004116 G
REG.3 = 010274          T$TSTM= 177777          T55     101136 G        WRD.52= 007066          $T1     025676
REG.4 = 010302          T$TSTS= 000000          T56     102240 G        WRD.53= 007074          $T10    032374
REG.5 = 010310          T$$HAR= 010002          T57     103462 G        WRD.54= 007104          $T11    033036
REG.6 = 010316          T$$HW = 010000          T58     104066 G        WRD.55= 007114          $T12    033302
REG.7 = 010324          T$$PRO= 010004          T59     104534 G        WRD.56= 007122          $T13    033702
REG.8 = 010332          T$$SOF= 010003          T6      030310 G        WRD.57= 007130          $T14    035044
REG.9 = 010340          T$$SW = 010001          T60     105066 G        WRD.58= 007140          $T15    036276
REM.TB  013434          T1      026064 G        T7      031140 G        WRD.59= 007152          $T16    037534
RE2     004144          T10     033022 G        T8      032124 G        WRD.6 = 006336          $T17    040424
RE3     004142          T11     033266 G        T9      032360 G        WRD.60= 007156          $T18    041620
RE4     004140          T12     033666 G        UAM   = 000200 G        WRD.61= 007174          $T19    042056
RH.ADD  013572          T13     035030 G        WRD.1 = 006276          WRD.62= 007210          $T2     026100
RH.ERR= 011060          T14     036262 G        WRD.10= 006406          WRD.63= 007216          $T20    043226
RH.TYP  013574          T15     037520 G        WRD.11= 006416          WRD.64= 007224          $T21    044166
RH.VEC  013576          T16     040410 G        WRD.12= 006426          WRD.65= 007232          $T22    044562
SC.SET= 010772          T17     041604 G        WRD.13= 006436          WRD.67= 007240          $T23    045230
SEV.FM= 006120          T18     042042 G        WRD.14= 006442          WRD.68= 007246          $T24    045566
SFPTBL  002376 G        T19     043212 G        WRD.15= 006454          WRD.69= 007252          $T25    046742
SIX.FM= 006100          T2      026330 G        WRD.16= 006462          WRD.7 = 006344          $T26    047534
STACK   012612          T20     044152 G        WRD.17= 006470          WRD.70= 007260          $T27    050660
STK.OF  012600          T21     044546 G        WRD.18= 006502          WRD.71= 007272          $T28    051436
STRIPP  014214          T22     045214 G        WRD.19= 006506          WRD.72= 007302          $T29    052422
SVCGBL= 177777          T23     045552 G        WRD.2 = 006302          WRD.73= 007310          $T3     026344
SVCINS= 177777          T24     046726 G        WRD.20= 006514          WRD.74= 007316          $T30    053534
SVCSUB= 177777          T25     047520 G        WRD.21= 006522          WRD.75= 007330          $T31    054330
SVCTAG= 177777          T26     050644 G        WRD.22= 006530          WRD.76= 007340          $T32    055760
SVCTST= 177777          T27     051422 G        WRD.23= 006544          WRD.77= 007346          $T33    057516
SYNC  = 010526          T28     052406 G        WRD.24= 006552          WRD.8 = 006360          $T34    060530
S$LSYM= 010000          T29     053520 G        WRD.25= 006560          WRD.9 = 006374          $T35    061464
TEN.FM= 006214          T3      026536 G        WRD.26= 006574          WRT.CS  017470          $T36    062116
THR.FM= 006034          T30     054314 G        WRD.27= 006602          WRT.DA  020024          $T37    063154
TIME.O= 011106          T31     055744 G        WRD.29= 006610          WRT.DS  022772          $T38    064062
TRBLE.= 011026          T32     057502 G        WRD.3 = 006314          WRT.D1  021576          $T39    065066
```

```
$T4     026552          $T45    071452          $T50    074216          $T55    100302          $T6     027666
$T40    065514          $T46    071636          $T51    074634          $T56    101152          $T60    104550
$T41    066502          $T47    072334          $T52    075706          $T57    102254          $T7     030324
$T42    067534          $T48    072626          $T53    076460          $T58    103476          $T8     031154
$T43    070306          $T49    073512          $T54    077260          $T59    104102          $T9     032140
$T44    071242          $T5     027244

. ABS.  105132      000
        000000      001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  34715 WORDS  ( 136 PAGES)
DYNAMIC MEMORY:  21558 WORDS  ( 82 PAGES)
ELAPSED TIME:  00:22:27
ML11,ML11/-SP/CR:SYM=SVC/ML,CZMLAC.DOC,ML2.P11,B16PG1.P11,B16PG2.P11,B16PG3.P11,B16PG4.P11,B16MUL.P11,B16SAV.P11,ML3.P11,ML4.P11
```

ML11 CREATED BY MACRO ON 18-MAY-81 AT 08:24 PAGE 1
SYMBOL CROSS REFERENCE CREF SEQ 0498

SYMBOL	VALUE		REFERENCES								
ADR	= 000020	G	#91-2189								
ARR.DA	= 010570		#95-2389	196-8115	199-8364	205-8663	213-9050	249-11475	249-11538	249-11613	280-13453
			286-13725	292-14080	293-14101	293-14122	294-14143	294-14160	305-14767	348-17722	354-18046
			373-19296	445-24381							
ARR.IN	011562		#86-1880	*184-7232	*184-7252	263-12425	263-12466	264-12472	303-14653	307-15172	307-15181
			325-16206	326-16224	328-16371	329-16389	370-18904	370-18908			
ARR.16	011574		#86-1887	*184-7222	*184-7233	*184-7253	*184-7254	263-12467			
ASSEMB	= 000010		5-13	5-13							
ASYNC	= 010464		#95-2387	190-7689	194-7904	196-8097	199-8346	205-8643	206-8705	212-9037	219-9386
			220-9410	220-9430	223-9581	226-9758	227-9779	227-9795	231-10006	231-10021	232-10042
			232-10059	232-10075	233-10113	233-10132	234-10160	238-10379	239-10398	239-10415	239-10432
			239-10448	240-10486	240-10505	241-10533	243-10775	243-10793	243-10810	243-10827	243-10846
			243-10880	243-10902	243-10926	246-11140	246-11156	246-11175	247-11196	247-11213	247-11230
			251-11791	255-12025	256-12046	256-12064	257-12100	257-12133	258-12179	262-12374	262-12390
			262-12407	263-12440	264-12486	266-12620	266-12636	266-12652	266-12671	268-12831	269-12854
			269-12875	273-13058	299-14447	311-15457	320-15918	330-16473	371-18978	381-19817	382-19841
			383-19884	383-19907	383-19923	386-20070	425-23219	431-23660	442-24211		
A.CAL	013140		#86-1904	100-2890	*100-2891	*100-2892	102-3014	*405-21097	406-21127	*408-21340	408-21369
A.GEN	013146		#86-1907	102-2990	102-3014	*405-21094	406-21120	*408-21337	408-21359		
BIT0	= 000001	G	#91-2173								
BIT00	= 000001	G	#91-2163								
BIT01	= 000002	G	#91-2162								
BIT02	= 000004	G	#91-2161								
BIT03	= 000010	G	#91-2160								
BIT04	= 000020	G	#91-2159								
BIT05	= 000040	G	#91-2158								
BIT06	= 000100	G	#91-2157								
BIT07	= 000200	G	#91-2156								
BIT08	= 000400	G	#91-2155								
BIT09	= 001000	G	#91-2154								
BIT1	= 000002	G	#91-2172								
BIT10	= 002000	G	#91-2153								
BIT11	= 004000	G	#91-2152								
BIT12	= 010000	G	#91-2151								
BIT13	= 020000	G	#91-2150								
BIT14	= 040000	G	#91-2149								
BIT15	= 100000	G	#91-2148								
BIT2	= 000004	G	#91-2171								
BIT3	= 000010	G	#91-2170								
BIT4	= 000020	G	#91-2169								
BIT5	= 000040	G	#91-2168								
BIT6	= 000100	G	#91-2167								
BIT7	= 000200	G	#91-2166								
BIT8	= 000400	G	#91-2165								
BIT9	= 001000	G	#91-2164								
BL$DIV	003752	G	#38-274	103-3086	418-22541	421-22855	449-24564				
BL$GT1	003016	G	#10-143	354-18038	449-24563						
BL$GT2	003140	G	#16-153	99-2708	100-2862	104-3112	255-12019	257-12094	258-12173	278-13340	280-13423
			292-14034	305-14739	307-15131	311-15430	317-15774	319-15891	326-16249	329-16413	332-16767
			337-17113	347-17696	348-17709	373-19283	400-20845	418-22565	422-22905	449-24562	
BL$LAS	105122	G	#449-24530	449-24531	449-24535						
BL$MOD	003764	G	#39-304	104-3094	418-22580	421-22859	449-24564				

SYMBOL		VALUE		REFERENCES								
BL$MUL		003526	G	#35-128	425-23136	449-24564						
BL$PU1		003302	G	#22-140	405-21082	449-24563						
BL$PU2		003376	G	#29-183	102-2989	102-2998	102-3007	257-12127	348-17743	418-22596	422-22876	449-24563
BL$SHF		003776	G	10-155	10-158	10-162	10-166	11-191	17-175	17-178	17-182	17-186
				17-207	22-143	22-147	22-153	23-180	29-192	29-196	29-202	30-234
				#40-338								
BOE	=	000400	G	#91-2193								
B.CAL		013142		#86-1905	100-2896	*100-2897	*100-2898	102-3016	*405-21098	406-21128	*408-21341	408-21370
B.GEN		013150		#86-1908	102-2999	102-3016	*405-21095	406-21121	*408-21338	408-21360		
CAL.CR		014612		#100-2832	405-21100	408-21343						
CHIP.S		011570		#86-1884	*184-7231	*184-7251	446-24398					
C$AU	=	000052		#5-13								
C$AUTO	=	000061		#5-13								
C$BRK	=	000022		#5-13								
C$BSEG	=	000004		#5-13								
C$BSUB	=	000002		#5-13								
C$CEFG	=	000045		#5-13								
C$CLCK	=	000062		#5-13								
C$CLEA	=	000012		#5-13								
C$CLOS	=	000035		#5-13								
C$CLP1	=	000006		#5-13								
C$CVEC	=	000036		#5-13								
C$DCLN	=	000044		#5-13								
C$DODU	=	000051		#5-13								
C$DRPT	=	000024		#5-13								
C$DU	=	000053		#5-13								
C$EDIT	=	000003		#5-13	5-65							
C$ERDF	=	000055		#5-13								
C$ERHR	=	000056		#5-13								
C$ERRO	=	000060		#5-13								
C$ERSF	=	000054		#5-13								
C$ERSO	=	000057		#5-13								
C$ESCA	=	000010		#5-13								
C$ESEG	=	000005		#5-13								
C$ESUB	=	000003		#5-13								
C$ETST	=	000001		#5-13								
C$EXIT	=	000032		#5-13								
C$GETB	=	000026		#5-13								
C$GETW	=	000027		#5-13								
C$GMAN	=	000043		#5-13								
C$GPHR	=	000042		#5-13								
C$GPLO	=	000030		#5-13								
C$GPRI	=	000040		#5-13								
C$INIT	=	000011		#5-13								
C$INLP	=	000020		#5-13								
C$MANI	=	000050		#5-13								
C$MEM	=	000031		#5-13								
C$MSG	=	000023		#5-13								
C$OPEN	=	000034		#5-13								
C$PNTB	=	000014		#5-13								
C$PNTF	=	000017		#5-13								
C$PNTS	=	000016		#5-13								

```
SYMBOL  VALUE          REFERENCES
C$PNTX  = 000015       #5-13
C$QIO   = 000377       #5-13
C$RDBU  = 000007       #5-13
C$REFG  = 000047       #5-13
C$RESE  = 000033       #5-13      #5-13
C$REVI  = 000003       #5-13       5-65
C$RFLA  = 000021       #5-13
C$RPT   = 000025       #5-13
C$SEFG  = 000046       #5-13
C$SPRI  = 000041       #5-13
C$SVEC  = 000037       #5-13
C$TPRI  = 000013       #5-13
DATA.L  = 010732       #95-2392
DAT.DM    014162       #98-2558    99-2665    292-14061  310-15368  310-15389  332-16688  332-16727  335-17036  336-17072
                        347-17657  373-19200  373-19240  418-22543  422-22883
DFPTBL    002356  G    #5-119
DIAGMC  = 000000        5-13       5-13
DIVMOD    003570       #36-196     38-275     39-305
DRIVE.    013132       #86-1898   *184-7228  *184-7248   206-8701   206-8714
DT.1      013534       #90-2117    425-23138  425-23140  425-23156  425-23263
DUMPER    024060  G    #180-6841   185-7398   187-7537   190-7690   193-7846   194-7905   196-8098   196-8107   196-8116
                        199-8347   199-8356   200-8369   205-8644   205-8655   205-8664   206-8706   212-9038   213-9051
                        219-9387   220-9411   220-9431   223-9582   226-9759   227-9780   227-9796   231-10007  232-10026
                        232-10043  232-10060  232-10076  233-10114  233-10119  233-10133  234-10161  234-10166  238-10380
                        239-10399  239-10416  239-10433  239-10449  240-10487  240-10492  241-10510  241-10534  241-10539
                        241-10560  243-10776  243-10794  243-10811  243-10828  243-10847  243-10881  243-10886  243-10903
                        243-10927  243-10932  243-10956  246-11141  246-11157  246-11176  247-11197  247-11214  247-11231
                        249-11476  249-11539  249-11614  251-11792  256-12030  256-12047  256-12065  257-12101  257-12134
                        258-12180  262-12375  262-12391  262-12408  263-12441  264-12487  266-12621  266-12637  266-12653
                        266-12672  269-12836  269-12855  269-12876  273-13059  273-13074  279-13381  280-13454  286-13726
                        292-14081  293-14102  293-14123  294-14144  294-14161  299-14448  305-14768  307-15187  311-15458
                        318-15802  320-15919  328-16351  330-16474  332-16800  332-16825  337-17127  341-17344  348-17723
                        354-18047  359-18341  360-18377  361-18416  365-18652  370-18914  371-18979  373-19297  377-19557
                        377-19579  377-19598  381-19818  382-19842  383-19885  383-19908  383-19924  386-20071  389-20194
                        392-20368  392-20403  395-20570  401-20864  406-21111  408-21350  412-21612  412-21635  413-21702
                        414-21732  416-21986  416-22010  417-22273  417-22297  418-22635  423-22936  425-23220  425-23238
                        425-23292  430-23606  431-23661  433-23765  437-23968  442-24207  442-24212  445-24382
D1.TEM  = 011152       #95-2397   *99-2692   *299-14416 *304-14721  307-15063 *307-15116 *311-15416 *317-15760 *319-15877
                       *332-16750 *336-17095 *347-17679 *373-19269 *400-20837 *418-22597  418-22603 *421-22850  422-22880
D2.TEM  = 011154       #95-2398   *99-2693   *299-14417 *304-14722  307-15064 *307-15117 *311-15417 *317-15761 *319-15878
                       *332-16751 *336-17096 *347-17680 *373-19270 *400-20838 *418-22598  418-22604 *421-22851  422-22881
EF.CON  = 000036  G    #91-2176
EF.NEW  = 000035  G    #91-2177
EF.PWR  = 000034  G    #91-2178
EF.RES  = 000037  G    #91-2175
EF.STA  = 000040  G    #91-2174
EIG.FM  = 006142       #92-2238
ELV.FM  = 006244       #92-2241
ERRBLK    002160  G    #5-96
ERRMSG    002156  G    #5-96
ERRNBR    002154  G    #5-96
ERRTYP    002152  G    #5-96
```

SYMBOL	VALUE		REFERENCES
ERR.CH	015212		#101-2973 405-21101 408-21344
EVL	= 000004	G	#91-2187
E$END	= 002100		#5-13
E$LOAD	= 000035		#5-13 5-65
E2.TEM	= 011156		#95-2399 *99-2694 *299-14418 *304-14723 307-15065 *307-15118 *311-15418 *317-15762 *319-15879 *332-16752 *336-17097 *347-17681 *373-19271 *400-20839 *418-22599 418-22605 *421-22830 *421-22831 422-22882
FIND.C	015430		#103-3081 412-21605 413-21695
FIRST.	016040		#109-3337 231-10000 238-10373 243-10769 268-12823 284-13653 298-14368 298-14390 318-15832
FIV.FM	= 006062		#92-2235 187-7543 205-8670 227-9769 227-9786 227-9802 231-10013 232-10032 232-10049 232-10066 233-10086 233-10125 234-10143 234-10172 238-10386 239-10405 239-10422 239-10439 240-10459 240-10498 241-10516 241-10545 243-10785 243-10800 243-10817 243-10834 243-10853 243-10895 243-10909 243-10938 246-11148 246-11163 247-11186 247-11204 247-11221 249-11620 256-12053 257-12107 258-12144 262-12381 262-12398 263-12418 263-12447 264-12493 266-12627 266-12643 266-12662 266-12678 269-12842 269-12861 269-12882 299-14454 312-15468 337-17133 365-18658 370-18920 378-19608 383-19914 386-20077 414-21738 416-22016 417-22306
FMT.1	= 004222		#92-2205 195-7919 425-23248 433-23775 437-23982 442-24225
FMT.10	= 004676		#92-2214 279-13394
FMT.11	= 004750		#92-2215 187-7548
FMT.12	= 005004		#92-2216 269-12847 269-12866 269-12887
FMT.13	= 005034		#92-2217 281-13490
FMT.14	= 005110		#92-2218 330-16485
FMT.15	= 005150		#92-2219 360-18389
FMT.16	= 005216		#92-2220 196-8137 200-8387 206-8681 213-9066 249-11494 249-11557 249-11630
FMT.17	= 005306		#92-2221 184-7282
FMT.18	= 005342		#92-2222
FMT.19	= 005412		#92-2223 406-21122 408-21364 425-23265
FMT.2	= 004266		#92-2206 194-7868 206-8715 223-9598 286-13745 341-17363 366-18673 431-23675
FMT.20	= 005462		#92-2224 406-21129 408-21371
FMT.21	= 005534		#92-2225 412-21622 412-21646 414-21716 414-21744 416-21997 416-22025 417-22284 417-22312
FMT.22	= 005606		#92-2226 418-22648 423-22946
FMT.23	= 005652		#92-2227 181-6903
FMT.24	= 005722		#92-2228 181-6910 181-6917 181-6924 181-6931 181-6938 182-6949 182-6956 182-6963 182-6970 182-6977 182-6984 182-6991 183-7003 183-7012 183-7019
FMT.25	= 005746		#92-2229 216-9225
FMT.26	= 005764		#92-2230 216-9235
FMT.3	= 004354		#92-2207
FMT.4	= 004400		#92-2208 251-11802
FMT.5	= 004430		#92-2209 337-17140 360-18360 360-18396 361-18431 401-20883
FMT.6	= 004532		#92-2210 373-19306
FMT.7	= 004562		#92-2211 258-12192 294-14177 382-19831 382-19851 383-19894
FMT.8	= 004610		#92-2212 446-24408 446-24419
FMT.9	= 004644		#92-2213 263-12452 264-12498 305-14777 371-18990 446-24394
FNC.1	= 007726		#94-2340 223-9584
FNC.10	= 010050		#94-2349 262-12377 262-12393 262-12410 263-12443 264-12489
FNC.11	= 010060		#94-2350 262-12394 266-12623 266-12639 266-12655 266-12674
FNC.12	= 010100		#94-2351 269-12838 269-12857 269-12878
FNC.13	= 010112		#94-2352 318-15811 370-18919 377-19560 377-19582 377-19599
FNC.14	= 010122		#94-2353 299-14449
FNC.15	= 010136		#94-2354 307-15188 328-16356 370-18924
FNC.16	= 010150		#94-2355 311-15460
FNC.17	= 010162		#94-2356 328-16354 330-16477 370-18918 371-18983

SYMBOL	VALUE	REFERENCES								
FNC.18	= 010172	#94-2357	348-17729	354-18053						
FNC.19	= 010204	#94-2358	389-20195							
FNC.2	= 007742	#94-2341	226-9761	227-9782	227-9798	256-12067	257-12103			
FNC.21	= 010216	#94-2359	279-13387							
FNC.22	= 010230	#94-2360	318-15803							
FNC.23	= 010242	#94-2361	185-7400	205-8566	249-11616					
FNC.3	= 007750	#94-2342	193-7850	194-7908	246-11144	247-11200	247-11217	248-11241	391-20338	395-20571
FNC.4	= 007756	#94-2343	231-10009	232-10062	233-10082	233-10121	234-10139	234-10168	396-20580	
FNC.5	= 007774	#94-2344	196-8120	200-8373	238-10382	239-10435	240-10455	240-10494	241-10512	241-10541
		242-10566	249-11480	249-11543	332-16802	332-16827	337-17128	348-17725	360-18347	360-18379
		361-18418	395-20573							
FNC.6	= 010004	#94-2345	187-7538	196-8119	200-8372	243-10778	243-10830	243-10849	243-10888	243-10905
		243-10934	243-10958	249-11479	249-11542	279-13382	341-17346	354-18049	365-18656	395-20572
FNC.7	= 010012	#94-2346	246-11143	247-11199	247-11216	248-11240	425-23294			
FNC.8	= 010022	#94-2347	251-11793	251-11796	256-12031	256-12048				
FNC.9	= 010036	#94-2348								
FORCE.	015602	#105-3184	412-21598	413-21686	416-21973	417-22256				
FOR.EC	015760	#107-3279	430-23594	431-23637	432-23700	433-23746	441-24157			
FOR.FM	= 006046	#92-2234	251-11797	256-12035	256-12071	273-13064	273-13079	281-13477	286-13731	330-16479
		371-18984	377-19562	377-19584	383-19929	389-20199	392-20373	393-20412	401-20869	412-21640
		416-21991	417-22278	425-23228	425-23297	437-23973				
F$AU	= 000015	#5-13								
F$AUTO	= 000020	#5-13								
F$BGN	= 000040	#5-13	5-39	6-226	6-274	6-305	6-337			
F$CLEA	= 000007	#5-13								
F$DU	= 000016	#5-13								
F$END	= 000041	#5-13	5-13	5-13	5-13	5-13	5-13	5-13	5-13	5-13
		5-13	5-13	5-13	5-13	5-13	5-13	5-13	5-13	5-39
		6-247	6-288	6-337						
F$HARD	= 000004	#5-13	6-226	6-247						
F$HW	= 000013	#5-13	5-119	5-139						
F$INIT	= 000006	#5-13								
F$JMP	= 000050	#5-13								
F$MOD	= 000000	#5-13	5-39	6-337						
F$MSG	= 000011	#5-13								
F$PROT	= 000021	#5-13	6-305	6-311						
F$PWR	= 000017	#5-13								
F$RPT	= 000012	#5-13								
F$SEG	= 000003	#5-13								
F$SOFT	= 000005	#5-13	6-274	6-288						
F$SRV	= 000010	#5-13								
F$SUB	= 000002	#5-13								
F$SW	= 000014	#5-13	5-150	5-162						
F$TEST	= 000001	#5-13								
GD.BLK	016070	#110-3388	332-16714	336-17063	340-17310	341-17331	346-17640	353-17994	353-18011	359-18318
		365-18618	365-18635	373-19228	376-19512	376-19534	392-20359	392-20394	393-20421	395-20551
		395-20561	400-20809	400-20814	405-21090	408-21333	411-21571	412-21599	413-21687	416-21926
		416-21974	417-22209	417-22261	418-22522	418-22622	421-22845	422-22919	425-23160	436-23932
		437-23947	441-24160	441-24171						
GOOD.B	011564	#86-1881	98-2560	110-3390	*184-7221	*307-15162				
G$CNTO	= 000200	#5-13								
G$DELM	= 000372	#5-13								

```
SYMBOL  VALUE            REFERENCES
G$DISP  = 000003         #5-13
G$EXCP  = 000400         #5-13
G$HILI  = 000002         #5-13
G$LOLI  = 000001         #5-13
G$NO    = 000000         #5-13
G$OFFS  = 000400         #5-13      6-239     6-240     6-241     6-242     6-243     6-244     6-245     6-285
G$OFSI  = 000376         #5-13      6-239     6-240     6-241     6-242     6-243     6-244     6-245     6-285
G$PRMA  = 000001         #5-13      6-239
G$PRMD  = 000002         #5-13      6-240     6-241     6-242     6-244
G$PRML  = 000000         #5-13      6-243     6-245     6-285
G$RADA  = 000140         #5-13
G$RADB  = 000000         #5-13
G$RADD  = 000040         #5-13      6-242
G$RADL  = 000120         #5-13      6-243     6-245     6-285
G$RADO  = 000020         #5-13      6-239     6-240     6-241     6-244
G$XFER  = 000004         #5-13
G$YES   = 000010         #5-13      6-239     6-240     6-241     6-242     6-243     6-244     6-245     6-285
HELP    = 000000         #5-4       5-8       5-30      5-48      5-67      5-105     5-121     5-152     #6-167
                          6-228     6-249     6-276     6-291     6-313     6-330     6-340
HOE     = 100000  G      #92-2204
HW.OR.    011160         #86-1874  *285-13672 286-13716
I8E     = 010000  G      #92-2201
IDU     = 000040  G      #91-2190
IER     = 020000  G      #92-2202
INTER   = 010672         #95-2391   193-7845   241-10559  243-10955  279-13380  307-15186  318-15801  328-16350  332-16824
                          370-18913
IO.BUF    011600         #86-1889    98-2561   104-3098  *104-3099   109-3340   110-3391   111-3441   304-14673  304-14697
                          307-15037  307-15089  326-16223  329-16388 *332-16712 *336-17062 *340-17309 *341-17327  341-17338
                          346-17596  352-17952 *353-18002  354-18032 *359-18312 *359-18313 *364-18608 *364-18609 *365-18626
                          365-18643 *369-18866  369-18879  369-18888 *370-18943  370-18946 *371-18961  371-18966 *373-19190
                         *373-19194 *373-19195 *373-19196 *376-19511 *376-19526  377-19552  377-19574  381-19808  386-20064
                          388-20180 *392-20356 *392-20378 *393-20420 *395-20550 *405-21061 *405-21084 *406-21142 *408-21329
                         *411-21570 *411-21588  412-21630 *413-21680  414-21727 *416-21925 *416-21967  416-22005 *417-22208
                         *417-22250  417-22292 *418-22513  418-22630 *421-22841  422-22927  430-23597  431-23651  432-23702
                          433-23748 *436-23931 *441-24159  445-24372
ISR     = 000100  G      #91-2191
IXE     = 004000  G      #92-2200
I$AU    = 000041         #5-13
I$AUTO  = 000041         #5-13
I$CLN   = 000041         #5-13
I$DU    = 000041         #5-13
I$HRD   = 000041         #6-226     #6-247
I$INIT  = 000041         #5-13
I$MOD   = 000041         #5-13       5-39      #5-39      6-337     #6-337
I$MSG   = 000041         #5-13
I$PROT  = 000040         #5-13      #6-305
I$PTAB  = 000041         #5-13
I$PWR   = 000041         #5-13
I$RPT   = 000041         #5-13
I$SEG   = 000041         #5-13
I$SETU  = 000041         #5-13
I$SFT   = 000041         #6-274     #6-288
```

```
SYMBOL  VALUE            REFERENCES
I$SRV   = 000041         #5-13
I$SUB   = 000041         #5-13
I$TST   = 000041         #5-13
J$JMP   = 000167         #5-13
LAST.B    016122         #111-3438   316-15710  316-15732  319-15854  382-19870
LAU       004210         #48-187     48-205
LAUTO     004164         #46-85      46-103
LCLEAN    105102         #448-24486  448-24506
LDU       004176         #47-135     47-153
LD.LNG    017132         #122-3973   307-15055  418-22602  422-22879
LINIT     024722         #184-7160   184-7310
LOAD.S    013604         #95-2407    99-2721
LOE     = 040000  G      #92-2203
LOT     = 000010  G      #91-2188
LRPT      004152         #45-32      45-50
LST.AR    011576         #86-1888   *184-7223  *184-7238  *184-7239  *184-7264  *184-7265   263-12424  263-12465  303-14652
                          307-15171  307-15180  370-18903  370-18907  430-23621
LST.BL    011572         #86-1886    111-3440  *184-7224  *184-7244  *184-7245  *184-7246  *184-7272  *184-7273  *184-7274
                          318-15819  381-19795  382-19874  383-19893  386-20054  388-20181  430-23595  431-23644  432-23703
                          432-23720  433-23743  445-24361
LST.DU    013134         #86-1900    183-7007  *184-7278  *184-7280   187-7517
L$ACP     002110  G      #5-65
L$APT     002036  G      #5-65
L$AU      004212  G       5-65       #48-205
L$AUT     002070  G      #5-65
L$AUTO    004166  G       5-65       #46-103
L$CCP     002106  G      #5-65
L$CLEA    105112  G       5-65       #448-24505
L$CO      002032  G      #5-65
L$DEPO    002011  G      #5-65
L$DESC    002130  G       5-65       #5-87
L$DESP    002076  G      #5-65
L$DEVP    002060  G      #5-65
L$DISP    002164  G       5-65       #5-103
L$DLY     002116  G      #5-65        99-2642    99-2669    99-2682    185-7386   190-7678   194-7885   226-9747   233-10093
                          240-10466  243-10860  246-11129  272-13043  278-13319  285-13663  285-13687  285-13700  298-14373
                          298-14394  299-14409  304-14679  304-14702  304-14714  307-15070  307-15093  307-15106  310-15373
                          310-15393  311-15409  316-15715  317-15740  317-15753  319-15858  319-15870  326-16229  329-16394
                          332-16693  332-16731  332-16743  335-17041  336-17076  336-17088  347-17661  347-17672  373-19208
                          373-19244  373-19262  400-20819  400-20830  418-22547  418-22609  422-22887
L$DTP     002040  G      #5-65
L$DTYP    002034  G      #5-65
L$DU      004200  G       5-65       #47-153
L$DUT     002072  G      #5-65
L$DVTY    002122  G       5-65       #5-80
L$EF      002052  G      #5-65
L$ENVI    002044  G      #5-65
L$ERRT    002152  G       5-65       #5-96
L$ETP     002102  G      #5-65
L$EXP1    002046  G      #5-65
L$EXP4    002064  G      #5-65
L$EXP5    002066  G      #5-65
```

SYMBOL	VALUE		REFERENCES							
L$HARD	002402	G	5-65	6-226	#6-226					
L$HIME	002120	G	#5-65							
L$HPCP	002016	G	#5-65							
L$HPTP	002022	G	#5-65							
L$HW	002356	G	5-65	5-119	#5-119					
L$ICP	002104	G	#5-65							
L$INIT	025666	G	5-65	#184-7310						
L$LADP	002026	G	#5-65							
L$LAST	= 105126	G	5-65	#449-24535						
L$LOAD	002100	G	#5-65							
L$LUN	002074	G	#5-65							
L$MREV	002050	G	#5-65							
L$NAME	002000	G	#5-65							
L$PRIO	002042	G	#5-65							
L$PROT	002750	G	5-65	#6-305						
L$PRT	002112	G	#5-65							
L$REPP	002062	G	#5-65							
L$REV	002010	G	#5-65							
L$RPT	004154	G	5-65	#45-50						
L$SOFT	002720	G	5-65	6-274	#6-274					
L$SPC	002056	G	#5-65							
L$SPCP	002020	G	#5-65							
L$SPTP	002024	G	#5-65							
L$STA	002030	G	#5-65							
L$SW	002376	G	5-65	5-150	#5-150					
L$TEST	002114	G	#5-65							
L$TIML	002014	G	#5-65							
L$UNIT	002012	G	#5-65	184-7153	184-7170	184-7205				
L10000	002374		5-119	#5-139						
L10001	002400		5-150	#5-162						
L10002	002476		6-226	#6-247						
L10003	002726		6-274	#6-288						
MEM.AR	= 010630		#95-2390							
ML.DUT	013602		#91-2145	99-2660	99-2730	158-5822	159-5847	162-6012	163-6037	167-6225	168-6253
			*184-7216	184-7287	185-7375	187-7527	190-7671	193-7834	193-7839	193-7841	194-7861
			194-7867	194-7876	194-7895	194-7900	195-7918	196-8072	199-8324	204-8600	204-8618
			210-8915	219-9376	220-9401	220-9421	223-9563	226-9739	231-9995	233-10105	234-10152
			238-10368	240-10478	241-10525	243-10764	243-10872	243-10918	245-11116	249-11454	249-11520
			249-11587	251-11775	255-12001	255-12008	255-12015	256-12079	257-12090	257-12116	257-12123
			258-12158	258-12162	258-12169	262-12364	263-12430	264-12476	266-12611	268-12818	272-13025
			277-13292	284-13647	285-13678	291-14009	298-14360	298-14385	303-14660	304-14691	307-15030
			307-15082	310-15359	310-15385	316-15700	316-15727	318-15826	318-15845	326-16215	329-16381
			332-16680	332-16708	332-16720	335-17029	336-17057	336-17068	340-17304	340-17316	346-17636
			346-17645	352-17942	353-18007	353-18016	359-18303	364-18599	365-18631	369-18871	370-18937
			371-18956	373-19182	373-19220	373-19233	376-19506	376-19519	381-19802	382-19865	386-20058
			388-20175	391-20347	392-20389	395-20545	395-20556	399-20776	405-21086	408-21305	411-21580
			413-21672	416-21957	417-22240	418-22518	418-22527	418-22618	421-22834	422-22915	425-23126
			425-23283	430-23586	430-23625	432-23692	432-23732	433-23755	436-23922	437-23943	441-24149
			441-24167	445-24366							
ML.LUN	013600		#91-2144	*184-7168	*184-7169	184-7170	184-7173	*184-7200	*184-7201	184-7205	184-7208
			184-7281	185-7406	188-7566	190-7703	195-7931	196-8156	200-8410	206-8722	213-9089
			224-9613	227-9812	234-10184	242-10578	243-10971	248-11253	249-11650	252-11821	281-13502

SYMBOL	VALUE	REFERENCES								
		286-13760	294-14192	300-14462	306-14793	307-15195	312-15472	320-15941	332-16815	338-17163
		342-17381	349-17766	354-18077	362-18461	366-18688	370-18931	371-18994	373-19335	384-19944
		406-21152	408-21385							
ML.REG	013154	#86-1910	98-2559	98-2560	98-2561	98-2562	99-2659	99-2662	99-2663	99-2666
		99-2678	99-2679	99-2692	99-2693	99-2694	99-2729	99-2732	99-2733	106-3204
		106-3208	106-3209	106-3212	106-3213	106-3216	106-3217	107-3280	107-3283	108-3288
		108-3290	108-3291	108-3292	109-3338	109-3339	109-3340	109-3341	110-3389	110-3390
		110-3391	110-3392	111-3439	111-3440	111-3441	111-3442	124-4099	124-4101	124-4102
		124-4104	126-4198	126-4200	126-4201	126-4204	126-4205	127-4258	127-4260	127-4261
		127-4263	129-4358	129-4360	129-4361	129-4364	129-4365	130-4418	130-4420	130-4421
		130-4423	132-4517	132-4519	132-4520	132-4523	132-4524	133-4577	133-4579	133-4580
		133-4582	135-4677	135-4679	135-4680	135-4683	135-4684	136-4730	136-4736	136-4738
		136-4739	136-4741	136-4742	138-4832	138-4838	138-4840	138-4841	138-4844	138-4845
		138-4850	139-4892	139-4898	139-4900	139-4901	139-4903	139-4904	141-4994	141-5000
		141-5002	141-5003	141-5006	141-5007	141-5012	142-5067	*142-5068	142-5074	142-5076
		142-5077	143-5083	*143-5084	143-5085	145-5179	*145-5180	145-5186	145-5188	145-5189
		145-5192	145-5193	*145-5198	145-5199	146-5254	146-5255	146-5261	146-5263	147-5268
		147-5270	147-5271	147-5272	147-5273	149-5367	149-5368	149-5374	149-5376	149-5377
		149-5380	149-5381	149-5386	149-5387	152-5520	155-5655	156-5700	156-5706	156-5708
		156-5709	156-5711	156-5712	158-5815	158-5816	158-5817	158-5818	158-5821	158-5824
		158-5825	158-5826	158-5832	158-5834	159-5839	159-5842	159-5846	159-5849	159-5850
		160-5892	160-5898	160-5900	160-5901	160-5903	160-5904	162-6005	162-6006	162-6007
		162-6008	162-6011	162-6014	162-6015	162-6016	162-6022	162-6024	162-6025	163-6032
		163-6036	163-6039	163-6040	164-6096	*164-6097	164-6103	164-6105	164-6106	165-6112
		*165-6113	165-6114	*167-6217	167-6218	167-6219	167-6220	167-6221	167-6224	167-6227
		167-6228	167-6229	167-6235	168-6241	168-6242	168-6245	168-6246	*168-6251	168-6252
		168-6255	168-6256	171-6399	171-6401	171-6402	171-6405	171-6406	181-6907	181-6908
		181-6914	181-6915	181-6921	181-6922	181-6928	181-6929	181-6935	181-6936	182-6946
		182-6947	182-6953	182-6954	182-6960	182-6961	182-6967	182-6968	182-6974	182-6975
		182-6981	182-6982	182-6988	182-6989	183-7000	183-7001	183-7009	183-7010	183-7016
		183-7017	*184-7191	184-7286	184-7289	184-7290	185-7374	185-7377	185-7378	185-7379
		185-7380	185-7393	187-7526	187-7529	187-7530	187-7531	187-7532	187-7547	190-7670
		190-7673	190-7674	190-7675	190-7685	193-7833	193-7836	193-7837	193-7839	193-7840
		194-7880	194-7881	194-7882	194-7897	194-7898	194-7899	196-8071	196-8074	196-8075
		196-8136	196-8323	199-8326	199-8327	200-8386	*204-8593	204-8599	204-8602	204-8603
		204-8614	204-8615	204-8617	204-8620	205-8625	205-8676	*206-8700	206-8701	210-8914
		210-8917	210-8918	213-9065	216-9197	216-9204	216-9207	216-9213	216-9223	219-9375
		219-9378	219-9379	219-9380	219-9381	219-9382	220-9400	220-9403	220-9404	220-9405
		220-9406	220-9420	220-9423	220-9424	220-9425	220-9426	223-9562	223-9565	223-9566
		223-9570	223-9589	226-9738	226-9741	226-9742	226-9744	226-9754	227-9775	227-9791
		231-9994	231-9997	231-9998	231-10001	231-10002	231-10017	232-10038	232-10055	232-10071
		233-10100	233-10102	233-10104	233-10107	233-10108	233-10109	234-10149	234-10151	234-10154
		234-10155	234-10156	238-10367	238-10370	238-10371	238-10374	238-10375	238-10390	239-10411
		239-10428	239-10444	240-10473	240-10475	240-10477	240-10480	240-10481	240-10482	241-10522
		241-10524	241-10527	241-10528	241-10529	241-10555	243-10763	243-10766	243-10767	243-10770
		243-10771	243-10789	243-10806	243-10823	243-10842	243-10867	243-10869	243-10871	243-10874
		243-10875	243-10876	243-10915	243-10917	243-10920	243-10921	243-10922	243-10951	245-11115
		245-11118	245-11119	245-11121	245-11122	246-11136	246-11152	246-11171	247-11192	247-11209
		247-11226	249-11453	249-11456	249-11460	249-11493	249-11519	249-11522	249-11523	249-11556
		249-11586	249-11589	249-11590	249-11601	249-11602	249-11629	251-11774	251-11777	251-11778
		251-11786	251-11787	255-12000	255-12003	255-12004	255-12005	255-12006	255-12007	256-12042
		256-12059	256-12060	256-12078	257-12113	257-12114	257-12128	257-12129	258-12154	258-12155

SYMBOL VALUE REFERENCES

258-12160 258-12161 262-12363 262-12366 262-12367 262-12368 262-12369 262-12370 262-12386
262-12403 263-12429 263-12432 263-12433 263-12434 263-12435 263-12436 264-12475 264-12478
264-12479 264-12480 264-12481 264-12482 266-12610 266-12613 266-12614 266-12615 266-12616
266-12632 266-12648 266-12667 268-12817 268-12820 268-12821 268-12824 268-12825 268-12827
269-12871 272-13024 272-13027 272-13028 272-13029 272-13036 272-13038 272-13040 272-13050
273-13069 277-13291 277-13294 277-13295 278-13301 278-13316 278-13326 284-13646 284-13649
284-13650 284-13652 284-13654 284-13655 285-13672 285-13673 285-13677 285-13680 285-13681
285-13682 285-13684 285-13694 285-13697 285-13707 291-14008 291-14011 291-14012 292-14062
292-14064 292-14065 292-14066 292-14067 298-14359 298-14362 298-14363 298-14364 298-14365
298-14366 298-14367 298-14369 298-14380 298-14384 298-14387 298-14388 298-14389 298-14391
299-14406 299-14416 299-14417 299-14418 303-14659 303-14662 303-14663 303-14664 303-14665
303-14666 303-14667 303-14668 304-14673 304-14674 304-14675 304-14686 304-14690 304-14693
304-14694 304-14695 304-14696 304-14697 304-14698 304-14699 304-14710 304-14711 304-14721
304-14722 304-14723 307-15029 307-15032 307-15033 307-15034 307-15035 307-15036 307-15037
307-15063 307-15064 307-15065 307-15066 307-15077 307-15081 307-15084 307-15085 307-15086
307-15087 307-15088 307-15089 307-15090 307-15102 307-15103 307-15116 307-15117 307-15118
310-15358 310-15361 310-15362 310-15365 310-15366 310-15367 310-15369 310-15380 310-15384
310-15387 310-15388 310-15390 310-15401 310-15402 311-15416 311-15417 311-15418 316-15699
316-15702 316-15703 316-15706 316-15707 316-15708 316-15709 316-15711 316-15722 316-15726
316-15729 316-15730 316-15731 316-15733 317-15749 317-15750 317-15760 317-15761 317-15762
318-15821 318-15822 318-15823 318-15825 318-15828 318-15829 318-15831 318-15833 318-15837
318-15844 319-15851 319-15852 319-15853 319-15855 319-15866 319-15867 319-15877 319-15878
319-15879 326-16214 326-16217 326-16218 326-16221 326-16222 326-16223 326-16225 326-16226
326-16236 327-16271 327-16279 327-16280 327-16285 327-16294 327-16295 327-16299 327-16307
327-16308 327-16313 328-16326 328-16327 328-16332 328-16333 328-16340 329-16380 329-16383
329-16384 329-16386 329-16387 329-16388 329-16390 329-16391 329-16401 329-16417 329-16430
330-16436 330-16438 330-16440 330-16442 330-16444 330-16450 330-16452 330-16458 330-16466
332-16679 332-16682 332-16683 332-16685 332-16686 332-16687 332-16689 332-16700 332-16707
332-16710 332-16711 332-16713 332-16715 332-16716 332-16719 332-16722 332-16723 332-16728
332-16739 332-16740 332-16750 332-16751 332-16752 335-17028 335-17031 335-17032 335-17033
335-17034 335-17035 335-17037 336-17052 336-17056 336-17059 336-17060 336-17061 336-17064
336-17065 336-17067 336-17070 336-17071 336-17073 336-17084 336-17085 336-17095 336-17096
336-17097 340-17303 340-17306 340-17307 340-17308 340-17311 340-17312 340-17315 341-17322
341-17323 341-17332 341-17333 346-17635 346-17638 346-17639 346-17641 346-17642 346-17644
346-17647 347-17652 347-17658 347-17668 347-17669 347-17679 347-17680 347-17681 352-17941
352-17944 352-17945 353-17995 353-17996 353-18006 353-18009 353-18010 353-18012 353-18013
353-18015 353-18018 353-18019 359-18302 359-18305 359-18306 359-18319 359-18320 364-18598
364-18601 364-18602 365-18619 365-18620 365-18630 365-18633 365-18634 365-18636 365-18637
369-18870 369-18873 369-18874 369-18875 369-18876 369-18878 369-18879 369-18880 369-18881
369-18882 369-18883 369-18885 369-18887 369-18888 369-18889 369-18890 370-18895 370-18896
370-18898 370-18936 370-18939 370-18940 370-18941 370-18942 370-18944 370-18945 370-18946
371-18951 371-18952 371-18953 371-18955 371-18958 371-18959 371-18960 371-18962 371-18965
371-18966 371-18969 371-18970 371-18971 371-18972 371-18974 371-18988 373-19181 373-19184
373-19185 373-19197 373-19198 373-19199 373-19201 373-19215 373-19219 373-19222 373-19223
373-19227 373-19229 373-19230 373-19232 373-19235 373-19236 373-19241 373-19255 373-19256
373-19269 373-19270 373-19271 376-19505 376-19508 376-19509 376-19510 376-19513 376-19514
376-19518 376-19521 376-19522 376-19532 376-19533 376-19535 376-19540 376-19541 377-19593
381-19801 381-19804 381-19805 381-19806 381-19807 381-19808 381-19809 381-19810 381-19811
381-19813 382-19837 382-19864 382-19867 382-19868 382-19869 382-19871 382-19872 382-19876
383-19900 383-19902 383-19903 386-20057 386-20060 386-20061 386-20062 386-20063 386-20064
386-20065 386-20066 388-20174 388-20177 388-20178 388-20179 388-20180 388-20181 389-20186
389-20187 389-20189 391-20346 391-20349 392-20354 392-20355 392-20360 392-20361 392-20363

SYMBOL	VALUE	REFERENCES								
		392-20379	392-20388	392-20391	392-20392	392-20393	392-20395	392-20396	392-20398	393-20422
		393-20423	395-20544	395-20547	395-20548	395-20549	395-20552	395-20553	395-20555	395-20558
		395-20559	395-20560	395-20562	395-20563	395-20565	399-20775	399-20778	399-20779	399-20780
		399-20791	400-20796	400-20799	400-20800	400-20803	400-20804	400-20807	400-20810	400-20811
		400-20813	400-20815	400-20826	400-20827	400-20837	400-20838	400-20839	405-21085	405-21088
		405-21089	405-21091	405-21092	408-21304	408-21310	408-21311	408-21334	408-21335	411-21569
		411-21572	411-21573	411-21579	411-21582	411-21583	412-21600	412-21601	412-21602	413-21671
		413-21674	413-21675	413-21688	413-21689	413-21690	416-21924	416-21927	416-21928	416-21956
		416-21959	416-21960	416-21975	416-21976	416-21977	416-21979	416-21981	417-22207	417-22210
		417-22211	417-22239	417-22242	417-22243	417-22257	417-22259	417-22260	417-22262	417-22263
		417-22264	417-22266	417-22268	418-22517	418-22520	418-22521	418-22523	418-22524	418-22526
		418-22529	418-22530	418-22533	418-22534	418-22538	418-22544	418-22559	418-22574	418-22603
		418-22604	418-22605	418-22606	418-22617	418-22620	418-22621	418-22623	418-22624	418-22625
		421-22833	421-22836	421-22837	421-22846	421-22847	422-22880	422-22881	422-22882	422-22884
		422-22899	422-22910	422-22914	422-22917	422-22918	422-22920	422-22921	422-22922	425-23125
		425-23128	425-23129	425-23144	425-23145	425-23150	425-23151	425-23154	425-23155	425-23158
		425-23159	425-23161	425-23162	425-23163	425-23170	425-23177	425-23184	425-23191	425-23198
		425-23209	425-23246	425-23282	425-23285	425-23286	425-23287	425-23302	430-23585	430-23588
		430-23589	430-23595	430-23596	430-23597	430-23598	430-23599	430-23601	430-23624	431-23631
		431-23632	431-23645	431-23648	431-23650	431-23651	431-23652	431-23653	431-23655	432-23691
		432-23694	432-23695	432-23701	432-23702	432-23703	432-23704	432-23705	432-23709	432-23725
		432-23731	432-23734	432-23735	433-23743	433-23747	433-23748	433-23749	433-23750	433-23754
		433-23757	433-23758	433-23759	436-23921	436-23924	436-23925	436-23926	436-23927	436-23928
		436-23929	436-23930	436-23933	436-23934	436-23938	437-23945	437-23946	437-23950	437-23953
		437-23955	437-23956	437-23960	437-23978	441-24148	441-24151	441-24152	441-24158	441-24161
		441-24162	441-2 66	441-24169	441-24170	441-24178	441-24179	441-24181	441-24182	441-24185
		441-24186	441-24188	441-24189	441-24193	442-24221	445-24365	445-24368	445-24369	445-24370
		445-24371	445-24372	445-24373	445-24374	445-24375	445-24377	448-24486		
MSGH1	002476	6-239	#6-256							
MSGH2	002512	6-240	#6-257							
MSGH3	002540	6-241	#6-258							
MSGH4	002563	6-242	#6-259							
MSGH5	002614	6-243	#6-260							
MSGH6	002646	6-244	#6-261							
MSGH7	002672	6-245	#6-262							
MSGS1	002726	6-285	#6-297							
MSDUMP	024070	180-6841	#181-6897							
NIB.SA	011152	#86-1872	95-2397	95-2398	95-2399	96-2434	96-2440	96-2446	96-2452	96-2458
		96-2464	97-2478	97-2484	97-2494	97-2503	114-3596	114-3603	114-3614	115-3626
		115-3638	115-3645	115-3656	115-3664	115-3676	116-3688	119-3826	119-3829	119-3837
		119-3839	119-3841	119-3848	119-3854	119-3856	119-3862	119-3866	*122-3990	*122-3997
		*122-4001	*123-4013	*123-4014	*123-4017	*123-4024	*123-4028	*123-4036	*123-4037	*123-4041
		*123-4049	*123-4050	401-20874						
NIN.FM	= 006166	#92-2239								
ONEFIL	= 000001	#2-4	3-1308	4-1309	5-34	5-164	6-190	6-353		
ONE.FM	= 006016	#92-2231	181-6899	190-7692						
OP.NUM	011560	#86-1878	*184-7220	184-7234	184-7240	184-7255	184-7266	223-9568	223-9596	263-12463
		326-16211	328-16372							
OSAPTS	= 000001	#5-13	#5-46	5-65						
OSAU	= 000001	#5-13	#5-46	5-65						
OSBGNR	= 000001	#5-13	#5-46	5-65						
OSBGNS	= 000001	#5-13	#5-46	5-65						

SYMBOL	VALUE		REFERENCES									
O$DU	= 000001		#5-13	#5-46	5-65							
O$ERRT	= 000001		#5-13	#5-46	5-65							
O$GNSW	= 000001		#5-13	#5-46	5-65							
O$POIN	= 000001		#5-13	#5-46	5-46	5-65						
O$SETU	= 000001		#5-13	#5-46	5-65							
PAR.DI	011566		#86-1883	*184-7214	219-9371	391-20334						
PD.TEM	013120		#86-1892	*99-2678	99-2702	*304-14710	305-14733	*307-15102	307-15125	*310-15401	311-15424	
			*317-15749	317-15768	*319-15866	319-15885	*326-16236	326-16243	*329-16401	329-16407	*332-16739	
			332-16761	*336-17084	337-17107	*347-17668	347-17691	*373-19255	373-19277	399-20772	*400-20826	
			418-22558	*418-22559	422-22898	*422-22899						
PHR.1	= 007366		#94-2325	231-10011	232-10030	234-10141	238-10384	239-10403	241-10514	243-10780	243-10798	
			243-10907	247-11184	251-11795	256-12033	256-12050	263-12416	264-12491	266-12676	269-12859	
			273-13062	273-13077	293-14128	294-14149	294-14166	378-19606	383-19927	386-20075	389-20197	
			412-21613	413-21703								
PHR.10	= 007564		#94-2334	307-15189	328-16357	370-18925						
PHR.11	= 007576		#94-2335	425-23239	430-23607							
PHR.12	= 007614		#94-2336	418-22638								
PHR.13	= 007636		#94-2337	423-22939								
PHR.14	= 007662		#94-2338	181-6898								
PHR.15	= 007714		#94-2339	216-9224	216-9234							
PHR.2	= 007404		#94-2326	227-9767	232-10047	233-10123	234-10170	239-10420	240-10496	241-10543	243-10815	
			243-10890	243-10936	256-12069	257-12105	258-12142	262-12379	266-12625	269-12840	383-19912	
PHR.3	= 007422		#94-2327	190-7691								
PHR.4	= 007454		#94-2328	185-7399	193-7852	196-8122	200-8375	205-8669	206-8707	249-11482	249-11545	
			249-11619	286-13727	299-14452	312-15466	318-15803	332-16806	332-16831	360-18349	360-18381	
			361-18420	365-18653	431-23662	445-24383						
PHR.5	= 007472		#94-2329	187-7541	227-9784	227-9800	232-10044	232-10064	233-10084	239-10417	239-10437	
			240-10457	242-10568	243-10812	243-10832	243-10851	243-10960	246-11146	246-11158	246-11161	
			247-11202	247-11219	262-12396	263-12445	266-12641	266-12657	269-12880	293-14090	293-14107	
			381-19820	412-21636	442-24214							
PHR.6	= 007504		#94-2330	232-10027	239-10400	243-10795	246-11177	258-12185				
PHR.7	= 007516		#94-2331	258-12181								
PHR.8	= 007540		#94-2332	377-19561								
PHR.9	= 007552		#94-2333	377-19583								
PNT	= 001000	G	#91-2194									
PRI	= 002000	G	#92-2199									
PRI00	= 000000	G	#91-2186									
PRI01	= 000040	G	#91-2185									
PRI02	= 000100	G	#91-2184									
PRI03	= 000140	G	#91-2183									
PRI04	= 000200	G	#91-2182									
PRI05	= 000240	G	#91-2181									
PRI06	= 000300	G	#91-2180									
PRI07	= 000340	G	#91-2179									
PRSN	002376	G	#5-160	215-9183	216-9195							
PTBL.P	011556		#86-1876	*184-7175	184-7176	184-7178	184-7179	184-7181	*184-7210	184-7211	184-7213	
			184-7215	184-7217	184-7225	184-7275						
P.AAA	004222		#69-912	92-2205								
P.AAB	004266		#69-924	92-2206								
P.AAC	004354		#69-942	92-2207								
P.AAD	004400		#69-949	92-2208								
P.AAE	004430		#69-957	92-2209								

SYMBOL	VALUE	REFERENCES	
P.AAF	004532	#70-983	92-2210
P.AAG	004562	#70-991	92-2211
P.AAH	004610	#70-999	92-2212
P.AAI	004644	#70-1009	92-2213
P.AAJ	004676	#70-1018	92-2214
P.AAK	004750	#71-1036	92-2215
P.AAL	005004	#71-1046	92-2216
P.AAM	005034	#71-1054	92-2217
P.AAN	005110	#71-1069	92-2218
P.AAO	005150	#72-1084	92-2219
P.AAP	005216	#72-1097	92-2220
P.AAQ	005306	#72-1116	92-2221
P.AAR	005342	#72-1126	92-2222
P.AAS	005412	#73-1144	92-2223
P.AAT	005462	#73-1158	92-2224
P.AAU	005534	#73-1172	92-2225
P.AAV	005606	#73-1186	92-2226
P.AAW	005652	#74-1202	92-2227
P.AAX	005722	#74-1216	92-2228
P.AAY	005746	#74-1223	92-2229
P.AAZ	005764	#74-1228	92-2230
P.ABA	006016	#74-1237	92-2231
P.ABB	006024	#74-1239	92-2232
P.ABC	006034	#74-1242	92-2233
P.ABD	006046	#75-1250	92-2234
P.ABE	006062	#75-1254	92-2235
P.ABF	006100	#75-1259	92-2236
P.ABG	006120	#75-1265	92-2237
P.ABH	006142	#75-1271	92-2238
P.ABI	006166	#75-1278	92-2239
P.ABJ	006214	#75-1286	92-2240
P.ABK	006244	#75-1294	92-2241
P.ABL	006276	#76-1307	92-2242
P.ABM	006302	#76-1309	92-2243
P.ABN	006314	#76-1313	92-2244
P.ABO	006322	#76-1315	92-2245
P.ABP	006330	#76-1317	92-2246
P.ABQ	006336	#76-1319	92-2247
P.ABR	006344	#76-1321	92-2248
P.ABS	006360	#76-1325	92-2249
P.ABT	006374	#76-1329	92-2250
P.ABU	006406	#76-1333	93-2255
P.ABV	006416	#76-1336	93-2256
P.ABW	006426	#76-1339	93-2257
P.ABX	006436	#76-1342	93-2258
P.ABY	006442	#76-1344	93-2259
P.ABZ	006454	#76-1348	93-2260
P.ACA	006462	#76-1350	93-2261
P.ACB	006470	#76-1352	93-2262
P.ACC	006502	#77-1360	93-2263
P.ACD	006506	#77-1362	93-2264
P.ACE	006514	#77-1364	93-2265

SYMBOL	VALUE	REFERENCES	
P.ACF	006522	#77-1366	93-2266
P.ACG	006530	#77-1368	93-2267
P.ACH	006544	#77-1372	93-2268
P.ACI	006552	#77-1374	93-2269
P.ACJ	006560	#77-1376	93-2270
P.ACK	006574	#77-1380	93-2271
P.ACL	006602	#77-1382	93-2272
P.ACM	006610	#77-1384	93-2273
P.ACN	006624	#77-1388	93-2274
P.ACO	006632	#77-1390	93-2275
P.ACP	006646	#77-1394	93-2276
P.ACQ	006654	#77-1396	93-2277
P.ACR	006660	#77-1398	93-2278
P.ACS	006666	#77-1400	93-2279
P.ACT	006674	#77-1402	93-2280
P.ACU	006700	#77-1404	93-2281
P.ACV	006710	#77-1407	93-2282
P.ACW	006716	#77-1409	93-2283
P.ACX	006726	#78-1416	93-2284
P.ACY	006732	#78-1418	93-2285
P.ACZ	006746	#78-1422	93-2286
P.ADA	006754	#78-1424	93-2287
P.ADB	006764	#78-1427	93-2288
P.ADC	006772	#78-1429	93-2289
P.ADD	007004	#78-1433	93-2290
P.ADE	007016	#78-1437	93-2291
P.ADF	007026	#78-1440	93-2292
P.ADG	007036	#78-1443	93-2293
P.ADH	007046	#78-1446	93-2294
P.ADI	007054	#78-1448	93-2295
P.ADJ	007066	#78-1452	93-2296
P.ADK	007074	#78-1454	93-2297
P.ADL	007104	#78-1457	93-2298
P.ADM	007114	#78-1460	93-2299
P.ADN	007122	#78-1462	93-2300
P.ADO	007130	#78-1464	93-2301
P.ADP	007140	#78-1467	93-2302
P.ADQ	007152	#79-1475	93-2303
P.ADR	007156	#79-1477	93-2304
P.ADS	007174	#79-1482	93-2305
P.ADT	007210	#79-1486	93-2306
P.ADU	007216	#79-1488	94-2311
P.ADV	007224	#79-1490	94-2312
P.ADW	007232	#79-1492	94-2313
P.ADX	007240	#79-1494	94-2314
P.ADY	007246	#79-1496	94-2315
P.ADZ	007252	#79-1498	94-2316
P.AEA	007260	#79-1500	94-2317
P.AEB	007272	#79-1504	94-2318
P.AEC	007302	#79-1507	94-2319
P.AED	007310	#79-1509	94-2320
P.AEE	007316	#79-1511	94-2321

SYMBOL	VALUE	REFERENCES	
P.AEF	007330	#79-1515	94-2322
P.AEG	007340	#79-1518	94-2323
P.AEH	007346	#79-1520	94-2324
P.AEI	007366	#80-1530	94-2325
P.AEJ	007404	#80-1535	94-2326
P.AEK	007422	#80-1540	94-2327
P.AEL	007454	#80-1549	94-2328
P.AEM	007472	#80-1554	94-2329
P.AEN	007504	#80-1558	94-2330
P.AEO	007516	#80-1562	94-2331
P.AEP	007540	#80-1568	94-2332
P.AEQ	007552	#80-1572	94-2333
P.AER	007564	#80-1576	94-2334
P.AES	007576	#81-1584	94-2335
P.AET	007614	#81-1589	94-2336
P.AEU	007636	#81-1595	94-2337
P.AEV	007662	#81-1602	94-2338
P.AEW	007714	#81-1611	94-2339
P.AEX	007726	#81-1615	94-2340
P.AEY	007742	#81-1619	94-2341
P.AEZ	007750	#81-1621	94-2342
P.AFA	007756	#81-1623	94-2343
P.AFB	007774	#81-1628	94-2344
P.AFC	010004	#81-1631	94-2345
P.AFD	010012	#81-1633	94-2346
P.AFE	010022	#82-1640	94-2347
P.AFF	010036	#82-1644	94-2348
P.AFG	010050	#82-1648	94-2349
P.AFH	010060	#82-1651	94-2350
P.AFI	010100	#82-1657	94-2351
P.AFJ	010112	#82-1661	94-2352
P.AFK	010122	#82-1664	94-2353
P.AFL	010136	#82-1668	94-2354
P.AFM	010150	#82-1672	94-2355
P.AFN	010162	#82-1676	94-2356
P.AFO	010172	#82-1679	94-2357
P.AFP	010204	#82-1683	94-2358
P.AFQ	010216	#82-1687	94-2359
P.AFR	010230	#82-1691	94-2360
P.AFS	010242	#83-1699	94-2361
P.AFT	010256	#83-1703	94-2362
P.AFU	010266	#83-1706	95-2367
P.AFV	010274	#83-1708	95-2368
P.AFW	010302	#83-1710	95-2369
P.AFX	010310	#83-1712	95-2370
P.AFY	010316	#83-1714	95-2371
P.AFZ	010324	#83-1716	95-2372
P.AGA	010332	#83-1718	95-2373
P.AGB	010340	#83-1720	95-2374
P.AGC	010346	#83-1722	95-2375
P.AGD	010354	#83-1724	95-2376
P.AGE	010362	#83-1726	95-2377

SYMBOL		VALUE	REFERENCES							
P.AGF		010370	#83-1728	95-2378						
P.AGG		010376	#83-1730	95-2379						
P.AGH		010404	#83-1732	95-2380						
P.AGI		010412	#83-1734	95-2381						
P.AGJ		010420	#83-1736	95-2382						
P.AGK		010430	#83-1739	95-2383						
P.AGL		010436	#83-1741	95-2384						
P.AGM		010444	#83-1743	95-2385						
P.AGN		010454	#83-1746	95-2386						
P.AGO		010464	#84-1753	95-2387						
P.AGP		010526	#84-1765	95-2388						
P.AGQ		010570	#84-1777	95-2389						
P.AGR		010630	#84-1788	95-2390						
P.AGS		010672	#84-1800	95-2391						
P.AGT		010732	#85-1815	95-2392						
P.AGU		010772	#85-1826	95-2393						
P.AGV		011026	#85-1836	95-2394						
P.AGW		011060	#85-1845	95-2395						
P.AGX		011106	#85-1853	95-2396						
P.CAL		013144	#86-1906	100-2884	*100-2885	*100-2886	102-3012	*405-21099	406-21126	*408-21342
P.GEN		013152	#86-1909	102-2984	102-3012	*405-21096	406-21119	*408-21339	408-21358	408-21368
RAS.IN		013124	#86-1895	*184-7230	*184-7250	307-15164	369-18867	369-18877		
RD.CS1		017540	#126-4191	178-6746	212-8993					
RD.DA		020074	#132-4510	178-6756	212-9007					
RD.DAT		013130	#86-1897	*126-4206	126-4207	*129-4366	129-4367	*132-4525	132-4526	*135-4685
			*138-4846	138-4847	*141-5008	141-5009	*145-5194	145-5195	*149-5382	149-5383
			152-5521	*155-5655	155-5656	*159-5842	159-5843	*163-6032	163-6033	*168-6247
			*171-6407	171-6408	196-8127	200-8380	205-8674	213-9059	249-11487	249-11550
RD.DS		022774	#171-6392	179-6800						
RD.D1		021662	#158-5813	179-6805						
RD.D2		022174	#162-6003	179-6810						
RD.D3		022522	#167-6215	179-6815						
RD.EE		021536	#155-5653	178-6786						
RD.EL		021474	#152-5518	178-6791						
RD.ER		017716	#129-4351	178-6751	212-9000					
RD.E1		020652	#141-4992	178-6766	212-9021					
RD.E2		021072	#145-5177	178-6771	212-9028					
RD.MR		020252	#135-4670	178-6761						
RD.PA		020444	#138-4830	178-6776	212-9014					
RD.PD		021334	#149-5365	178-6781						
RD.REG		023470	#177-6719	196-8088	199-8337	205-8630	249-11470	249-11533	249-11608	
REG.IN		013136	#86-1902	158-5819	162-6009	167-6222	*249-11581	*249-11603	*249-11647	
REG.1	=	010256	#94-2362	181-6909						
REG.10	=	010346	#95-2375							
REG.11	=	010354	#95-2376							
REG.12	=	010362	#95-2377							
REG.13	=	010370	#95-2378							
REG.14	=	010376	#95-2379	182-6990	425-23241	425-23296				
REG.15	=	010404	#95-2380	183-7002	430-23609	431-23663	433-23768			
REG.16	=	010412	#95-2381							
REG.17	=	010420	#95-2382	181-6937	185-7401					
REG.18	=	010430	#95-2383	181-6916						

```
SYMBOL  VALUE          REFERENCES
REG.19  = 010436       #95-2384    181-6923
REG.2   = 010266       #95-2367    182-6948    256-12052
REG.20  = 010444       #95-2385    183-7011
REG.21  = 010454       #95-2386    183-7018
REG.3   = 010274       #95-2368    182-6955    247-11232
REG.4   = 010302       #95-2369    182-6969
REG.5   = 010310       #95-2370    182-6962    257-12135   258-12182
REG.6   = 010316       #95-2371    181-6930    193-7847    382-19845   383-19888   383-19909
REG.7   = 010324       #95-2372    182-6976    206-8708
REG.8   = 010332       #95-2373
REG.9   = 010340       #95-2374    182-6983
REM.TB    013434       #89-2080    100-2882    100-2888    100-2894    106-3205    106-3210    106-3214
RE2       004144        43-66     #43-96
RE3       004142        43-74     #43-95
RE4       004140        43-83     #43-94
RH.ADD    013572       #90-2137   *184-7178    184-7189
RH.ERR  = 011060       #95-2395    185-7397    187-7536
RH.TYP    013574       #90-2138   *184-7180
RH.VEC    013576       #91-2143   *184-7182
SC.SET  = 010772       #95-2393
SEV.FM  = 006120       #92-2237    328-16359   332-16808   341-17352
SFPTBL    002376  G    #5-150
SIX.FM  = 006100       #92-2236    193-7853    196-8123    200-8376    242-10571   243-10963   248-11242   249-11483   249-11546
                        258-12187   279-13388   280-13463   293-14092   293-14109   293-14130   294-14151   294-14168   332-16832
                        348-17730   354-18054   360-18352   360-18384   361-18423   396-20581
STACK     012612       #86-1891   *97-2497    *97-2498    *97-2510    *97-2511    100-2843    102-2983    359-18336   360-18358
                        360-18371   360-18394   361-18411   361-18429
STK.OF    012600       #86-1890    95-2410    *99-2655    99-2713    *99-2717
STRIPP    014214       #99-2649    100-2836    359-18324
SVCGBL  = 177777       #5-13      #5-22       5-65        5-65        5-65        5-65        5-65        5-65        5-65
                        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65
                        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65
                        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65
                        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65
                        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65
                        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65
                        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65
                        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65
                        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65
                        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-80
                        5-80        5-80        5-87        5-87        5-87        5-96        5-96        5-96        5-103
                        5-103       5-103       5-119       5-119       5-119       5-119       5-119       5-119       5-150
                        5-150       5-150       5-150       5-150       5-150       6-226       6-226       6-226       6-274
                        6-274       6-274       6-305       6-305       6-305
SVCINS  = 177777       #5-13      #5-19       5-65        5-65        5-65        5-65        5-65        5-65        5-65
                        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65
                        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65
                        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65        5-65
                        5-65        5-65        5-65        5-65        5-80        5-80        5-87        5-87        5-103
```

SYMBOL	VALUE		REFERENCES								
			5-103	5-103	5-103	5-103	5-103	5-103	5-103	5-103	5-103
			5-103	5-103	5-103	5-103	5-103	5-103	5-103	5-103	5-103
			5-103	5-103	5-103	5-103	5-103	5-103	5-103	5-103	5-103
			5-103	5-103	5-103	5-103	5-103	5-103	5-103	5-103	5-103
			5-103	5-103	5-103	5-103	5-103	5-103	5-103	5-103	5-103
			5-103	5-103	5-103	5-103	5-103	5-103	5-103	5-103	5-103
			5-103	5-103	5-103	5-103	5-103	5-103	5-119	5-150	6-226
			6-239	6-239	6-239	6-239	6-240	6-240	6-240	6-240	6-240
			6-241	6-241	6-241	6-241	6-241	6-242	6-242	6-242	6-242
			6-242	6-243	6-243	6-243	6-244	6-244	6-244	6-244	6-244
			6-245	6-245	6-245	6-247	6-274	6-285	6-285	6-285	6-288
SVCSUB	= 177777		#5-13	#5-21							
SVCTAG	= 177777		#5-13	#5-23	5-139	5-139	5-139	5-162	5-162	5-162	6-247
			6-247	6-247	6-288	6-288	6-288				
SVCTST	= 177777		#5-13	#5-20							
SYNC	= 010526		#95-2388	196-8106	199-8355	205-8654	233-10118	234-10165	240-10491	241-10538	243-10885
			243-10931	273-13073	332-16799	337-17126	341-17343	359-18340	360-18376	361-18415	365-18651
			377-19556	377-19578	377-19597	389-20193	392-20367	392-20402	401-20863	406-21110	408-21349
			412-21611	412-21634	413-21701	414-21731	416-21985	416-22009	417-22272	417-22296	418-22634
			423-22935	425-23237	425-23291	430-23605	433-23764	437-23967	442-24206		
S$LSYM	= 010000		#5-13	#5-139	#5-162	#6-247	#6-288				
TEN.FM	= 006214		#92-2240								
THR.FM	= 006034		#92-2233	185-7402	194-7909	213-9055	220-9395	220-9415	220-9435	305-14772	307-15191
			318-15806	320-15923	370-18927	373-19301	382-19826	382-19846	383-19889	391-20339	406-21115
			408-21354	412-21616	414-21710	418-22639	423-22940	425-23242	430-23610	433-23769	442-24216
TIME.O	= 011106		#95-2396								
TRBLE.	= 011026		#95-2394	395-20569							
TST.LN	016154		#114-3577	299-14424	305-14747	307-15139	311-15438	317-15782	319-15899	348-17717	373-19291
			401-20858								
TWO.FM	= 006024		#92-2232	206-8709	221-9451	223-9585	318-15812	431-23664	445-24385		
T$ARGC	= 000002		#5-65	5-65	#5-65	5-65	5-65	#5-65	5-65	5-65	#5-65
			5-65	5-65	#5-65	5-65	5-65	#5-65	5-65	5-65	
T$CODE	= 000130		#6-239	6-239	#6-239	6-239	#6-239	6-239	#6-240	6-240	#6-240
			6-240	#6-240	6-240	#6-241	6-241	#6-241	6-241	#6-241	6-241
			#6-242	6-242	#6-242	6-242	#6-242	6-242	#6-243	6-243	#6-243
			6-243	#6-243	6-243	#6-244	6-244	#6-244	6-244	#6-244	6-244
			#6-245	6-245	#6-245	6-245	#6-245	6-245	#6-285	6-285	#6-285
			6-285	#6-285	6-285						
T$ERRN	= 000000		#5-13								
T$EXCP	= 000000		#6-239	6-239	#6-240	6-240	#6-241	6-241	#6-242	6-242	#6-244
			6-244								
T$FREE	105126	G	449-24530	449-24531	#449-24532						
T$GMAN	= 000000		#5-13								
T$HILI	= 000007		#6-239	6-239	#6-240	6-240	#6-241	6-241	#6-242	6-242	#6-244
			6-244								
T$LAST	= 000000		#5-13								
T$LOLI	= 000000		#6-239	6-239	#6-240	6-240	#6-241	6-241	#6-242	6-242	#6-244
			6-244								
T$LSYM	= 010000		#5-13	5-13	5-139	5-162	6-247	6-288			
T$NEST	= 177777		#5-13	5-39	#5-39	5-39	5-119	#5-119	5-119	5-139	5-139
			5-139	#5-139	5-150	#5-150	5-150	5-162	5-162	5-162	#5-162
			6-226	#6-226	6-226	6-247	6-247	6-247	#6-247	6-274	#6-274

SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES								
			6-274	6-288	6-288	6-288	#6-288	6-305	#6-305	6-305	6-311
			6-311	6-311	#6-311	6-337	6-337	6-337	#6-337		
T$NS0	= 000000		#5-39	6-337							
T$NS1	= 000021		#5-119	5-139	#5-150	5-162	#6-226	6-247	#6-274	6-288	#6-305
			6-311								
T$PTHV	= 000000	G	5-65	#449-24536							
T$PTNU	= 000000		#5-13								
T$SAVL	= 177777		#5-13								
T$SEGL	= 177777		#5-13								
T$SUBN	= 000000		#5-13								
T$TAGL	= 177777		#5-13								
T$TAGN	= 010005		#5-13	5-119	5-119	#5-119	5-150	5-150	#5-150	6-226	6-226
			#6-226	6-274	6-274	#6-274	6-305	6-305	#6-305		
T$TEMP	= 000000		#5-103	5-103	5-103	#5-103	5-103	5-103	#5-103	5-103	5-103

SYMBOL	VALUE		REFERENCES	
T13	035030	G	5-103	#235-10210
T14	036262	G	5-103	#242-10600
T15	037520	G	5-103	#243-10993
T16	040410	G	5-103	#248-11275
T17	041604	G	5-103	#249-11672
T18	042042	G	5-103	#252-11842
T19	043212	G	5-103	#259-12228
T2	026330	G	5-103	#188-7587
T20	044152	G	5-103	#265-12531
T21	044546	G	5-103	#266-12701
T22	045214	G	5-103	#270-12924
T23	045552	G	5-103	#274-13113
T24	046726	G	5-103	#282-13528
T25	047520	G	5-103	#287-13786
T26	050644	G	5-103	#295-14218
T27	051422	G	5-103	#300-14485
T28	052406	G	5-103	#306-14819
T29	053520	G	5-103	#307-15218
T3	026536	G	5-103	#191-7727
T30	054314	G	5-103	#312-15495
T31	055744	G	5-103	#321-15967
T32	057502	G	5-103	#331-16520
T33	060514	G	5-103	#332-16856
T34	061450	G	5-103	#338-17190
T35	062102	G	5-103	#342-17408
T36	063140	G	5-103	#349-17788
T37	064046	G	5-103	#355-18103
T38	065052	G	5-103	#362-18483
T39	065500	G	5-103	#366-18709
T4	027230	G	5-103	#195-7953
T40	066466	G	5-103	#372-19020
T41	067520	G	5-103	#373-19357
T42	070272	G	5-103	#378-19638
T43	071226	G	5-103	#384-19965
T44	071436	G	5-103	#387-20110
T45	071622	G	5-103	#389-20222
T46	072320	G	5-103	#393-20447
T47	072612	G	5-103	#396-20608
T48	073476	G	5-103	#402-20921
T49	074202	G	5-103	#407-21178
T5	027652	G	5-103	#196-8178
T50	074620	G	5-103	#408-21406
T51	075672	G	5-103	#415-21781
T52	076444	G	5-103	#416-22062
T53	077244	G	5-103	#417-22350
T54	100266	G	5-103	#418-22687
T55	101136	G	5-103	#423-22981
T56	102240	G	5-103	#425-23327
T57	103462	G	5-103	#434-23811
T58	104066	G	5-103	#438-24015
T59	104534	G	5-103	#443-24259
T6	030310	G	5-103	#201-8436

SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES								
T60		163066	G	5-103	#447-24452						
T7		031140	G	5-103	#207-8748						
T8		032124	G	5-103	#214-9115						
T9		032360	G	5-103	#217-9263						
UAM	=	000200	G	#91-2192							
WRD.1	=	006276		#92-2242	227-9768	231-10012	232-10028	232-10045	233-10124	234-10171	238-10385	239-10401
				239-10418	240-10497	241-10544	243-10781	243-10796	243-10813	243-10891	243-10937	246-11147
				246-11159	246-11178	262-12380	266-12626	269-12841				
WRD.10	=	006406		#93-2255	213-9052	279-13386	305-14769	320-15920	330-16475	337-17130	341-17349	348-17727
				354-18051	373-19298	377-19558	377-19580	386-20072	396-20579	401-20865	416-21989	417-22276
				425-23226	433-23766	437-23971	442-24213					
WRD.11	=	006416		#93-2256	227-9766	227-9783	227-9799	233-10122	234-10140	234-10169	240-10495	241-10513
				241-10542	243-10889	243-10906	243-10935	246-11145	256-12032	256-12049	256-12068	257-12104
				257-12137	258-12184	262-12378	262-12395	266-12624	266-12640	266-12656	266-12675	269-12839
				269-12858	273-13061	273-13076	378-19605	383-19911	383-19926	386-20074	389-20196	
WRD.12	=	006426		#93-2257	187-7540	193-7851	196-8121	200-8374	205-8668	231-10010	232-10063	233-10083
				238-10383	239-10436	240-10456	242-10567	243-10779	243-10831	243-10850	243-10959	247-11201
				247-11218	249-11481	249-11544	249-11618	251-11794	262-12411	263-12444	264-12490	269-12879
				279-13385	299-14451	312-15465	328-16355	332-16804	332-16829	337-17129	341-17348	348-17726
				354-18050	360-18348	360-18380	361-18419	396-20578	414-21734	416-22012	417-22299	
WRD.13	=	006436		#93-2258	193-7848	247-11233	318-15804					
WRD.14	=	006442		#93-2259	194-7906	223-9583	247-11234	307-15190	328-16358	370-18926	371-18980	382-19843
				383-19886	425-23221	425-23293	437-23969					
WRD.15	=	006454		#93-2260	256-12034	256-12070	257-12106	258-12143	258-12186			
WRD.16	=	006462		#93-2261	256-12051	263-12417						
WRD.17	=	006470		#93-2262	257-12136	258-12183	383-19910					
WRD.18	=	006502		#93-2263	266-12677							
WRD.19	=	006506		#93-2264	226-9760	227-9781	227-9797	231-10008	232-10061	232-10077	233-10120	234-10138
				234-10167	238-10381	239-10434	240-10454	240-10493	241-10511	241-10540	241-10561	243-10777
				243-10829	243-10848	243-10887	243-10904	243-10933	243-10957	246-11142	247-11198	247-11215
				247-11235	256-12066	257-12102	262-12376	262-12392	262-12409	263-12442	264-12488	266-12622
				266-12638	266-12654	266-12673	269-12837	269-12856	269-12877	286-13728	332-16801	332-16826
				341-17345	348-17724	354-18048	359-18342	360-18378	361-18417	365-18655	377-19559	377-19581
				378-19604								
WRD.2	=	006302		#92-2243	232-10031	232-10048	234-10142	239-10404	239-10421	241-10515	243-10799	243-10816
				243-10908	246-11162	247-11185						
WRD.20	=	006514		#93-2265	242-10569	243-10961	273-13078					
WRD.21	=	006522		#93-2266	273-13060	273-13063	273-13075					
WRD.22	=	006530		#93-2267	299-14453	312-15467	332-16807					
WRD.23	=	006544		#93-2268	332-16803	332-16828	341-17347	360-18351	360-18383	361-18422	365-18657	392-20372
				393-20411								
WRD.24	=	006552		#93-2269	293-14086	293-14103	293-14124	294-14145	294-14162	337-17132	341-17351	401-20867
WRD.25	=	006560		#93-2270	337-17131	341-17350						
WRD.26	=	006574		#93-2271	389-20198							
WRD.27	=	006602		#93-2272	381-19821	383-19913	383-19925	383-19928				
WRD.29	=	006610		#93-2273	381-19819							
WRD.3	=	006314		#92-2244	227-9785	232-10065	239-10438	243-10833	247-11203	262-12397	266-12642	269-12860
WRD.30	=	006624		#93-2274	386-20073	386-20076						
WRD.31	=	006632		#93-2275	382-19844	383-19887						
WRD.32	=	006646		#93-2276	293-14089	293-14106	293-14127	294-14148	294-14165			
WRD.33	=	006654		#93-2277	293-14087	293-14104	293-14125	294-14146	294-14163			
WRD.34	=	006660		#93-2278	293-14091	293-14108	293-14129	294-14150	294-14167			

SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE	REFERENCES								
WRD.35	= 006666	#93-2279	279-13383	286-13730	305-14771	445-24384				
WRD.36	= 006674	#93-2280	286-13729							
WRD.37	= 006700	#93-2281	193-7849	194-7907	213-9053	305-14770	328-16353	330-16476	391-20336	
WRD.38	= 006710	#93-2282	213-9054							
WRD.39	= 006716	#93-2283	330-16478	360-18350	360-18382	361-18421	365-18654			
WRD.4	= 006322	#92-2245	227-9801	233-10085	240-10458	243-10852	247-11220	263-12446	264-12492	266-12661
		269-12881								
WRD.40	= 006726	#93-2284	280-13457	281-13473						
WRD.41	= 006732	#93-2285	373-19300							
WRD.42	= 006746	#93-2286	280-13458	281-13474						
WRD.43	= 006754	#93-2287	232-10029	232-10046	239-10402	239-10419	243-10797	243-10814	246-11160	246-11179
WRD.44	= 006764	#93-2288								
WRD.45	= 006772	#93-2289	279-13384	299-14450						
WRD.46	= 007004	#93-2290	280-13460	281-13476	373-19299					
WRD.47	= 007016	#93-2291	281-13475							
WRD.48	= 007026	#93-2292	311-15459							
WRD.49	= 007036	#93-2293								
WRD.5	= 006330	#92-2246	219-9390	293-14126	294-14147	294-14164	406-21114	408-21353		
WRD.50	= 007046	#93-2294	320-15922	348-17728	354-18052	370-18917	371-18982			
WRD.51	= 007054	#93-2295	280-13459	320-15921						
WRD.52	= 007066	#93-2296	187-7539	196-8118	200-8371	205-8667	249-11478	249-11541	249-11617	425-23240
		425-23295	430-23608							
WRD.53	= 007074	#93-2297	220-9445							
WRD.54	= 007104	#93-2298	280-13461							
WRD.55	= 007114	#93-2299	280-13462							
WRD.56	= 007122	#93-2300	196-8117	200-8370	205-8665	249-11477	249-11540	249-11615	318-15810	328-16352
		370-18915								
WRD.57	= 007130	#93-2301								
WRD.58	= 007140	#93-2302								
WRD.59	= 007152	#93-2303	378-19607							
WRD.6	= 006336	#92-2247	220-9414	220-9434	293-14088	293-14105	392-20371	393-20410		
WRD.60	= 007156	#93-2304	370-18916	371-18981						
WRD.61	= 007174	#93-2305	242-10570	243-10962	412-21639	414-21737	416-22015	417-22305		
WRD.62	= 007210	#93-2306	187-7542							
WRD.63	= 007216	#94-2311								
WRD.64	= 007224	#94-2312	401-20868	406-21113	408-21352					
WRD.65	= 007232	#94-2313								
WRD.67	= 007240	#94-2314	414-21733	416-21990	416-22011	417-22298	425-23227	437-23972		
WRD.68	= 007246	#94-2315								
WRD.69	= 007252	#94-2316	425-23225	437-23970						
WRD.7	= 006344	#92-2248	219-9389	220-9413	220-9433	220-9446	391-20337	392-20370	392-20405	
WRD.70	= 007260	#94-2317								
WRD.71	= 007272	#94-2318	433-23767							
WRD.72	= 007302	#94-2319	442-24215							
WRD.73	= 007310	#94-2320	401-20866							
WRD.74	= 007316	#94-2321	412-21615	412-21638	414-21709	414-21736	416-22014	417-22301		
WRD.75	= 007330	#94-2322	412-21614	412-21637	413-21704	414-21735	416-22013	417-22300		
WRD.76	= 007340	#94-2323	416-21988	417-22275	418-22637	423-22938				
WRD.77	= 007346	#94-2324	417-22277							
WRD.8	= 006360	#92-2249	220-9432	392-20404	406-21112	408-21351				
WRD.9	= 006374	#92-2250	219-9388	220-9412	332-16805	332-16830	392-20369	416-21987	417-22274	418-22636
		423-22937								

SYMBOL CROSS REFERENCE CREF

SYMBOL	VALUE		REFERENCES								
WRT.CS	017470		#124-4093	174-6547	210-8923	211-8955					
WRT.DA	020024		#130-4412	174-6557	210-8929	211-8963					
WRT.DS	022772		#169-6305	175-6601							
WRT.D1	021576		#156-5699	175-6606							
WRT.D2	022110		#160-5891	175-6611							
WRT.D3	022422		#164-6095	175-6616							
WRT.EE	021534		#153-5571	174-6587							
WRT.EL	021472		#150-5434	175-6596							
WRT.ER	017646		#127-4252	174-6552	210-8926	211-8959					
WRT.E1	020566		#139-4891	174-6567	211-8939	211-8971					
WRT.E2	020774		#142-5066	174-6572	211-8942						
WRT.MR	020202		#133-4571	174-6562							
WRT.PA	020360		#136-4729	174-6577	211-8936	211-8967					
WRT.PD	021226		#146-5253	174-6582	373-19226						
WRT.RE	023102		#173-6520	196-8083	199-8332	204-8608	249-11465	249-11528	249-11595		
WT.DAT	013126		#86-1896	*126-4203	126-4207	*129-4363	129-4367	*132-4522	132-4526	*135-4682	135-4686
			*138-4843	138-4847	*141-5005	141-5009	*145-5191	145-5195	*149-5379	149-5383	*152-5519
			152-5521	*155-5654	155-5656	*159-5841	159-5843	*163-6031	163-6033	*168-6244	168-6248
			*171-6404	171-6408	196-8128	200-8381	213-9060	249-11488	249-11551		
W.C.SI	013122		#86-1893	*184-7229	*184-7249	369-18878	369-18887	371-18967			
XOR.LN	016650		#118-3805	332-16776	337-17121						
X$ALWA	= 000000		#5-13								
X$FALS	= 000040		#5-13								
X$OFFS	= 000400		#5-13								
X$TRUE	= 000020		#5-13								
$END.L	105130	G	#449-24543								
$PATCH	002756	G	#6-327								
$SAVE2	004044	G	35-128	40-371	#43-61	103-3082	114-3578	124-4093	126-4191	127-4252	129-4351
			130-4412	132-4510	133-4571	135-4670	136-4729	138-4830	139-4891	141-4992	142-5066
			145-5177	146-5253	149-5365	156-5699	158-5813	160-5891	162-6003	164-6095	167-6215
			171-6392	173-6520	177-6719	226-9735	231-9991	238-10364	243-10760	245-11112	272-13020
			369-18865	395-20541	445-24360	449-24563					
$SAVE3	004060	G	10-143	11-191	16-153	17-207	29-183	30-234	#43-68	95-2408	101-2974
			105-3185	184-7160	216-9194	219-9370	223-9560	255-11999	262-12362	376-19504	436-23920
			441-24147	449-24562							
$SAVE4	004076	G	#43-76	118-3806	190-7666	196-8064	251-11773	381-19793	449-24562		
$SAVE5	004116	G	36-196	40-371	#43-85	99-2650	100-2832	187-7515	193-7830	199-8315	204-8591
			210-8911	249-11446	268-12809	277-13289	284-13643	291-14002	298-14353	303-14648	307-15022
			310-15355	316-15696	325-16203	332-16677	335-17021	340-17298	345-17584	352-17939	359-18299
			364-18597	373-19178	399-20765	405-21055	408-21300	411-21568	416-21923	417-22206	418-22506
			421-22828	425-23124	430-23582	449-24562					
$T1	025676		#185-7372	185-7431							
$T10	032374		#219-9370	221-9475							
$T11	033036		#223-9560	224-9635							
$T12	033302		#226-9735	228-9839							
$T13	033702		#231-9991	235-10211							
$T14	035044		#238-10364	242-10601							
$T15	036276		#243-10760	243-10994							
$T16	037534		#245-11112	248-11276							
$T17	040424		#249-11446	249-11673							
$T18	041620		#251-11773	252-11843							
$T19	042056		#255-11999	259-12229							

SYMBOL	VALUE	REFERENCES	
$T2	026100	#187-7515	188-7588
$T20	043226	#262-12362	265-12532
$T21	044166	#266-12610	266-12702
$T22	044562	#268-12809	270-12925
$T23	045230	#272-13020	274-13114
$T24	045566	#277-13289	282-13529
$T25	046742	#284-13643	287-13787
$T26	047534	#291-14002	295-14219
$T27	050660	#298-14353	300-14486
$T28	051436	#303-14648	306-14820
$T29	052422	#307-15022	307-15219
$T3	026344	#190-7666	191-7728
$T30	053534	#310-15355	312-15496
$T31	054330	#316-15696	321-15968
$T32	055760	#325-16203	331-16521
$T33	057516	#332-16677	332-16857
$T34	060530	#335-17021	338-17191
$T35	061464	#340-17298	342-17409
$T36	062116	#345-17584	349-17789
$T37	063154	#352-17939	355-18104
$T38	064062	#359-18299	362-18484
$T39	065066	#364-18597	366-18710
$T4	026552	#193-7830	195-7954
$T40	065514	#369-18865	372-19021
$T41	066502	#373-19178	373-19358
$T42	067534	#376-19504	378-19639
$T43	070306	#381-19793	384-19966
$T44	071242	#386-20053	387-20111
$T45	071452	#388-20174	389-20223
$T46	071636	#391-20333	393-20448
$T47	072334	#395-20541	396-20609
$T48	072626	#399-20765	402-20922
$T49	073512	#405-21055	407-21179
$T5	027244	#196-8064	196-8179
$T50	074216	#408-21300	408-21407
$T51	074634	#411-21568	415-21782
$T52	075706	#416-21923	416-22063
$T53	076460	#417-22206	417-22351
$T54	077260	#418-22506	418-22688
$T55	100302	#421-22828	423-22982
$T56	101152	#425-23124	425-23328
$T57	102254	#430-23582	434-23812
$T58	103476	#436-23920	438-24016
$T59	104102	#441-24147	443-24260
$T6	027666	#199-8315	201-8437
$T60	104550	#445-24360	447-24453
$T7	030324	#204-8591	207-8749
$T8	031154	#210-8911	214-9116